Zero Data Loss Recovery Appliance
Administrator's Guide



Release 23.1
F72078-11
May 2026

ORACLE

Zero Data Loss Recovery Appliance Administrator's Guide, Release 23.1

F72078-11

Copyright © 2014, 2026, Oracle and/or its affiliates.

Contributing Authors: Glenn Maxey, Terence Buencamino, Lance Ashdown, Aishwarya Minocha

Contributors: Andrew Babb, Donna Carver, Tim Chien, Sean Connolly, Donna Cooksey, Bill Fischer, Mahesh Girkar, Ray Guzman, Dah-Yoh Lim, Colin McGregor, Kant Patel, Chris Plakyda, Padmaja Potineni, Kathy Rich, Jony Safi, Toru Sasaki, Lawrence To, Randy Urbano, Steven Wertheimer

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software, software documentation, data (as defined in the Federal Acquisition Regulation), or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle®, Java, MySQL, and NetSuite are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Contents

Preface

Audience i
Documentation Accessibility i
Related Documents i
Conventions i

Changes in Zero Data Loss Recovery Appliance Administrator's Guide Release 23.1

1 Introduction to Recovery Appliance

1.1 Traditional Database Backup Techniques 1
1.1.1 Weekly Full and Daily Incremental Backups 1
1.1.2 Incremental Backups and RECOVER COPY 2
1.1.3 Full Backups to a Third-Party Deduplicating Appliance 3
1.1.4 Third-Party Storage Snapshots 4
1.2 Data Protection Challenges in the Modern Enterprise 5
1.3 Oracle's Recovery Appliance Solution 6
1.3.1 Elimination of Data Loss 8
1.3.1.1 Protection of Ongoing Transactions 8
1.3.1.2 Secure Replication 8
1.3.1.3 Autonomous Tape Archival 9
1.3.1.4 End-to-End Data Validation 12
1.3.2 Minimal Backup Overhead 13
1.3.2.1 Delta Push 14
1.3.2.2 Delta Store 15
1.3.3 Improved End-to-End Data Protection Visibility 16
1.3.4 Cloud-Scale Protection 17
1.3.4.1 Policy-Based Data Protection Management 17
1.3.4.2 Database-Aware Space Management 18
1.3.4.3 Scalable Architecture 18
1.3.5 Maximum Availability: Recovery Appliance with Oracle Data Guard 18

Copyright © 2014, 2026, Oracle and/or its affiliates.

1.4 What's Next? 19

2 Recovery Appliance Architecture

2.1 The Recovery Appliance Environment 1
2.1.1 Main Components of the Recovery Appliance Environment 2
2.1.2 User Accounts in the Recovery Appliance Environment 4
2.1.3 Lifecycle of a Backup: Scenario 6
2.2 Protected Databases 8
2.2.1 Recovery Appliance Backup Modules 8
2.2.2 Protection Policies 9
2.2.2.1 Protection Policy Attributes 10
2.2.2.2 Recovery Windows 12
2.2.2.3 Backup Polling Policies 13
2.2.3 Supported Oracle Database Releases 13
2.3 Real-Time Redo Transport 13
2.4 Recovery Appliance Metadata Database 14
2.4.1 Delta Store 15
2.4.2 Delta Pools 15
2.4.3 Automated Delta Pool Space Management 16
2.4.4 Recovery Appliance Schema 16
2.4.5 Recovery Appliance Catalog 17
2.5 Recovery Appliance Storage 17
2.5.1 Recovery Appliance Storage Locations 18
2.5.1.1 Benefits of Recovery Appliance Storage 18
2.5.1.2 Oracle ASM and Recovery Appliance Storage 18
2.5.1.3 DELTA Storage Location 19
2.5.2 Backup Polling Locations 19
2.5.2.1 Stages of Backup Polling 20
2.5.2.2 How Recovery Appliance Processes Backups in Backup Polling Directories 20
2.5.3 How Recovery Appliance Manages Storage Space 21
2.5.3.1 Recovery Window Goal 22
2.5.3.2 Reserved Space 22
2.5.3.3 Guaranteed Copy 23
2.5.3.4 Maximum Retention Window 24
2.5.3.5 Recover Window Compliance 25
2.5.3.6 Archival and Encrypted Backups 25
2.5.3.7 Space-Efficient Encrypted Backups 26
2.6 Ransomware and Cybersecurity 27
2.7 Oracle Secure Backup 28
2.7.1 Tape Archival 29

Copyright © 2014, 2026, Oracle and/or its affiliates.

2.7.2 Tape Retrieval 30
2.8 Recovery Appliance Replication 30
2.8.1 How a Downstream Recovery Appliance Processes Backups 30
2.8.2 Replication Use Cases 31
2.9 Data Encryption Techniques 34
2.9.1 Transparent Data Encryption (TDE) on Production Database Tablespaces 35
2.9.2 Redo Encryption Using LOG_ARCHIVE_DEST_n 36
2.9.3 Tape Drive-Based Hardware Encryption 37

Part I Managing Recovery Appliance

3 Recovery Appliance Workflow

3.1 Separation of Duties in Recovery Appliance Administration 1
3.2 Prerequisites for Recovery Appliance Administration 2
3.2.1 Tools for Recovery Appliance Administration 2
3.2.2 Planning for Recovery Appliance 2
3.3 Setup and Configuration for Recovery Appliance 4
3.4 Maintenance Tasks for Recovery Appliance 6

4 Getting Started with Cloud Control for Recovery Appliance

4.1 Displaying All Recovery Appliances in the Enterprise 1
4.2 Accessing the Recovery Appliance Home Page 2
4.3 Accessing the Recovery Appliance Storage Locations Page 4

5 Securing the Operations of the Recovery Appliance

5.1 Remote Handling of Recovery Appliance System Logs 2

6 TLS Overview and Configuration

6.1 Certificate Management 2
6.1.1 Using Your Organization's CA process for TLS Certificates 3
6.1.2 Manually Creating TLS Certificates with RACLI 6
6.2 Configuring TLS Data Security on the Recovery Appliance 8
6.3 Configuring TLS Data Security on the Client 11
6.4 Trouble-shooting TLS 13
6.5 TLS and Replication Setup 14

Copyright © 2014, 2026, Oracle and/or its affiliates.

7 Managing Protection Policies with Recovery Appliance

7.1 About Protection Policies 1
7.1.1 Purpose of Protection Policies 1
7.1.2 Overview of Protection Policies 2
7.1.3 Guidelines for Protection Policies 2
7.1.4 User Interfaces for Protection Policies 4
7.1.4.1 Accessing the Create Protection Policy Page in Cloud Control 4
7.1.4.2 DBMS_RA Procedures Relating to Protection Policies 5
7.1.4.3 Recovery Catalog Views for Protection Policies 5
7.1.5 Basic Tasks for Managing Protection Policies 6
7.2 Protection Policy Attributes 7
7.3 Creating a Protection Policy 9
7.4 Updating a Protection Policy 14
7.5 Deleting a Protection Policy 15
7.6 Creating a Backup Polling Policy (Command-Line Only) 16

8 Configuring Recovery Appliance for Protected Database Access

8.1 Overview of Protected Database Access 1
8.2 Basic Tasks for Configuring Protected Database Access 2
8.3 Creating Virtual Private Catalog Accounts 3
8.4 Enrolling Protected Databases 4
8.5 Updating Protected Database Properties 7
8.6 User Interfaces for Configuring Protected Database Access 9
8.6.1 DBMS_RA Procedures Relating to Protected Database Access 9
8.6.2 Recovery Catalog Views for Protected Database Access 10

9 Copying Backups to Tape with Recovery Appliance

9.1 About Copying Backups to Tape with Recovery Appliance 1
9.1.1 Purpose of Copying Backups to Tape with Recovery Appliance 1
9.1.2 Overview of Copying Backups to Tape with Recovery Appliance 2
9.1.2.1 About Tape Operations on Recovery Appliance 3
9.1.2.2 Grouping Backup Pieces 4
9.1.2.3 Recovery Appliance Components for Managing Tape Operations 4
9.1.2.4 Backup Retention on Tape 6
9.1.2.5 About Pausing and Resuming Tape Backup Operations 6
9.1.2.6 About Using Oracle Secure Backup with Recovery Appliance 7
9.1.3 User Interfaces for Recovery Appliance 9
9.1.3.1 Accessing Recovery Appliance in Cloud Control 9
9.1.3.2 Accessing Recovery Appliance Using DBMS_RA 9

Copyright © 2014, 2026, Oracle and/or its affiliates.

9.1.4 Basic Tasks for Copying Backups to Tape with Recovery Appliance 10
9.1.4.1 Accessing the Oracle Secure Backup Domain Using Cloud Control 12
9.2 Creating Tape Backup Job Components 13
9.2.1 Creating a Media Manager Library 13
9.2.2 Creating an Attribute Set 16
9.3 Managing Tape Backup Job Components 18
9.3.1 Managing a Media Manager Library Using Cloud Control 18
9.3.2 Managing an Attribute Set Using Cloud Control 20
9.3.3 Managing an SBT Library Using DBMS_RA 21
9.3.4 Managing an Attribute Set Using DBMS_RA 22
9.4 Creating a Tape Backup Job 23
9.5 Managing a Tape Backup Job 28
9.6 Scheduling a Tape Backup Job 30
9.7 Pausing and Resuming Tape Backup Operations 31
9.8 Viewing the Status of Tape Backup Operations 32
9.8.1 Viewing the Status of Tape Backup Operations Using Cloud Control 33
9.8.1.1 Viewing the Media Manager Library Status 33
9.8.1.2 Viewing the Tape Backup Job Status 33
9.8.2 Viewing the Status of Tape Backup Operations Using DBMS_RA 33
9.8.2.1 Checking the SBT Library Status 33
9.8.2.2 Checking the Tape Backup Job Status 34
9.8.2.3 Reviewing SBT Job Runs Using DBMS_RA 35
9.8.2.4 Checking the Status of Oracle Scheduler Jobs 35

10 Converting Tape Configurations to Cloud

10.1 About Tape to Cloud Conversion 1
10.2 Requirements for Converting Tape Configurations to Cloud 2
10.3 Converting a Tape Configuration to Cloud 2

11 Archiving Backups to Cloud

11.1 Grouping Backup Pieces 1
11.2 Pre-requisites for Archive-to-Cloud 2
11.3 Flow for Archive-to-Cloud Storage 2
11.4 Oracle Key Vault and Recovery Appliance 3
11.4.1 Review: Oracle Key Vault 4
11.4.2 Creating the Endpoints 4
11.4.3 Creating the Endpoint Group 5
11.4.4 Creating a Wallet 5
11.4.5 Associating Default Wallet with Endpoints 6
11.4.6 Acquiring the Enrollment Tokens 7

Copyright © 2014, 2026, Oracle and/or its affiliates.

11.4.7 Downloading the OKV Client Software 8
11.5 Recovery Appliance Cloud Archive Configuration 9
11.5.1 Configuring the Credential Wallet and Encryption Keystore 9
11.5.2 Installing the OKV Client Software 10
11.5.3 Enabling the Encryption Keystore and Creating a TDE Master Key 11
11.5.4 Creating Cloud Objects for Archive-to-Cloud 11
11.5.5 Adding Cloud Location 13
11.5.6 Adding an Immutable Cloud Location 16
11.6 Configuring the ZFS OCI Object Storage as a Cloud Repository 17
11.7 Creating a Job Template 17
11.8 Creating or Re-Creating Protected Database TDE Master Keys 19

12 Implementing Immutable Backups

12.1 Managing Recovery Window Compliance 2
12.2 Managing Compliance Holds 6

13 Archival Backups

13.1 Managing Archival Backups 1

14 Recovery Appliance Partners

15 Replicating Backups with Recovery Appliance

15.1 Overview of Recovery Appliance Replication 1
15.1.1 Protection Policies for Replication 2
15.1.2 How Recovery Appliance Replicates Backups: Basic Process 2
15.1.3 Archived Backup Mirroring 3
15.1.4 How RMAN Restores Backups in a Replication Environment 4
15.2 Additonal Replication Topology Examples 5
15.2.1 Replication to One Downstream Recovery Appliance 5
15.2.2 Replication to Multiple Downstream Recovery Appliances 6
15.2.3 Replication Using Different Policies on Downstream Recovery Appliances 7
15.2.4 Cascaded Replication 8
15.2.5 Bi-Directional Replication between Recovery Appliances 9
15.2.6 Replication Request-Only Mode 10
15.2.7 Read-Only Replication between Recovery Appliances 11
15.2.8 Copy All Backups Replication 11
15.3 About Replication Partners and Users Accounts 12
15.4 RACLI Commands Relating to Replication 13

Copyright © 2014, 2026, Oracle and/or its affiliates.

15.4.1 Guidance for Replication Configuration 14
15.4.2 Configuring Replication between Recovery Appliances Using RACLI 15
15.4.3 Converting Existing Replication Settings to RACLI Management 21
15.4.4 Configuring Replication with a Virtual Air Gap 21
15.5 Accessing the Replication Page in Cloud Control 23
15.5.1 Configuring Recovery Appliance Replication Using Cloud Control 24
15.6 DBMS_RA Procedures Relating to Replication 28
15.6.1 Configuring Recovery Appliance for Replication Using DBMS_RA 28
15.6.1.1 Assumptions for the Replication Examples 29
15.6.1.2 Configuring a Downstream Recovery Appliance for Replication Using DBMS_RA 30
15.6.1.3 Configuring an Upstream Recovery Appliance for Replication Using DBMS_RA 33
15.6.1.4 Configuring a Protected Database for Recovery Appliance Replication 40
15.6.1.5 Testing a Recovery Appliance Replication Server Configuration 40
15.6.2 Configuring Recovery Appliance Replication with TLS Using DBMS_RA 42
15.6.2.1 Case 1: One-Way Replication; TLS disabled on Downstream 42
15.6.2.2 Case 2: One-Way Replication; TLS enabled on Downstream 42
15.6.2.3 Case 3: Two-Way Replication; TLS disabled on Downstream 45
15.6.2.4 Case 4: Two-Way Replication; TLS enabled on Downstream 48
15.6.2.5 Case 5: Two-Way Replication; TLS disabled on Upstream 51
15.6.2.6 Case 6: Two-Way Replication; TLS disabled on Upstream and Downstream 53
15.6.3 Recovery Catalog Views for Replication 53

16 Implementing Additional High Availability Strategies

16.1 Managing Temporary Outages with a Backup and Redo Failover Strategy 1
16.1.1 Overview of the Backup and Redo Failover Feature 1
16.1.2 Configuring Backup and Redo Failover 2
16.1.2.1 Configuring the Primary Recovery Appliance for Backup and Redo Failover 2
16.1.2.2 Configuring the Alternate Recovery Appliance for Backup and Redo Failover 3
16.1.2.3 Configuring Replication for Backup and Redo Failover 4
16.1.2.4 Configuring the Protected Database for Backup and Redo Failover 7
16.2 Implementing DR Failover to Downstream Recovery Appliance 8
16.2.1 Setup and Configuration for Failover 9
16.2.1.1 Creating VPC Users 9
16.2.1.2 Modifying Configuration for Transport Failover 11
16.2.1.3 Configuring the Replication Server 12
16.2.1.4 Configuring Upstream and Downstream Recovery Appliances 13
16.2.1.5 Registering the Protected Database on the Upstream Recovery Appliance 14

Copyright © 2014, 2026, Oracle and/or its affiliates.

16.2.1.6 Adding Remaining Grants to the Upstream and Downstream Recovery Appliance 17
16.2.1.7 Configuring Channel Device Parameters 17
16.2.2 Configuring Upstream and Downstream Recovery Appliance 18
16.2.3 Backup Operation 20
16.2.4 Real-Time Redo Transport 21
16.2.4.1 Configuring the VPC User for Real-Time Redo Transport 21
16.2.4.2 Option 1: Use Data Guard Broker to Configure Real-Time Redo Transport 21
16.2.4.3 Option 2: Use log_archive* Parameters to Configure Real-Time Redo Transport 23
16.3 Replication Mode for HADR 23
16.4 Backup Anywhere Mode for Data Guard 24
16.5 Request_Only Mode for Data Guard 25
16.6 Replication Read-Only Mode when Migrating to New Data Center 25

17 Monitoring the Recovery Appliance

17.1 About Monitoring the Recovery Appliance 1
17.1.1 Purpose of Monitoring the Recovery Appliance 1
17.1.2 Overview of Recovery Appliance Monitoring Capabilities 1
17.1.2.1 Cloud Control 1
17.1.2.2 Oracle Configuration Manager 2
17.1.2.3 Auto Service Request (ASR) 2
17.1.3 Cloud Control Interface for Monitoring the Recovery Appliance 2
17.1.4 Basic Tasks for Monitoring the Recovery Appliance 4
17.2 Modifying the Metric and Collection Settings 4
17.3 Viewing the Incident Manager Page 5
17.4 Monitoring Performance 7
17.4.1 Generating Performance Statistics by Using the rastat Utility 7
17.4.1.1 Prerequisites for Running the rastat Utility 7
17.4.1.2 Running the rastat Utility 7
17.4.2 Testing Network Throughput 9

18 Accessing Recovery Appliance Reports

18.1 Purpose of Recovery Appliance Reports 1
18.2 Pre-Created Oracle Analytics Publisher Reports 2
18.3 Accessing the Recovery Appliance Reports Page in Cloud Control 4
18.4 Oracle Analytics Publisher Report Scheduling 5
18.5 Basic Tasks for Accessing Recovery Appliance Reports 6

Copyright © 2014, 2026, Oracle and/or its affiliates.

Part II Recovery Appliance Life-Cycle

19 Best Practices for the Recovery Appliance

20 Storage Management for the Recovery Appliance

21 Running Recovery Appliance Checks

22 Updating the Recovery Appliance

23 Parameter Update Rollback

24 Migrating Backups to other Recovery Appliances

24.1 Configuring Migration between Recovery Appliances Using RACLI 2

Part III Recovery Appliance Reference

25 DBMS_RA Package Reference

25.1 ABORT 5
25.2 ABORT_RECOVERY_APPLIANCE 5
25.3 ADD_DB 6
25.4 ADD_REPLICATION_SERVER 7
25.5 CONFIG 8
25.6 COPY_BACKUP 10
25.7 COPY_BACKUP_PIECE 11
25.8 CREATE_ARCHIVAL_BACKUP 13
25.9 CREATE_POLLING_POLICY 16
25.10 CREATE_PROTECTION_POLICY 16
25.11 CREATE_RESTORE_POINT 21
25.12 CREATE_REPLICATION_SERVER 22
25.13 CREATE_SBT_ATTRIBUTE_SET 23

Copyright © 2014, 2026, Oracle and/or its affiliates.

25.14 CREATE_SBT_JOB_TEMPLATE 24
25.15 CREATE_SBT_JOB_TEMPLATE 27
25.16 CREATE_SBT_LIBRARY 27
25.17 DELETE_DB 28
25.18 DELETE_POLLING_POLICY 29
25.19 DELETE_PROTECTION_POLICY 30
25.20 DELETE_REPLICATION_SERVER 30
25.21 DELETE_SBT_ATTRIBUTE_SET 31
25.22 DELETE_SBT_JOB_TEMPLATE 31
25.23 DELETE_SBT_LIBRARY 31
25.24 ESTIMATE_SPACE 32
25.25 GET_REDO_TRANSPORT_LAG 33
25.26 GRANT_DB_ACCESS 33
25.27 KEY_REKEY 33
25.28 KEY_REKEY 34
25.29 KEY_REKEY 34
25.30 MIGRATE_TAPE_BACKUP 35
25.31 MOVE_BACKUP 35
25.32 MOVE_BACKUP_PIECE 37
25.33 PAUSE_REPLICATION_DATABASE 39
25.34 PAUSE_REPLICATION_SERVER 39
25.35 PAUSE_SBT_LIBRARY 40
25.36 POPULATE_BACKUP_PIECE 40
25.37 QUEUE_SBT_BACKUP_TASK 41
25.38 REMOVE_REPLICATION_SERVER 42
25.39 RENAME_DB 42
25.40 RESET_ERROR 43
25.41 RESUME_DB 43
25.42 RESUME_REPLICATION_DATABASE 44
25.43 RESUME_REPLICATION_SERVER 44
25.44 RESUME_SBT_LIBRARY 45
25.45 REVOKE_DB_ACCESS 45
25.46 SET_SYSTEM_DESCRIPTION 46
25.47 SHUTDOWN 46
25.48 SHUTDOWN_RECOVERY_APPLIANCE 47
25.49 STARTUP 47
25.50 STARTUP_RECOVERY_APPLIANCE 48
25.51 SUSPEND_DB 48
25.52 UPDATE_ARCHIVAL_BACKUP_KEEP 49
25.53 UPDATE_DB 50
25.54 UPDATE_POLLING_POLICY 51
25.55 UPDATE_PROTECTION_POLICY 51

Copyright © 2014, 2026, Oracle and/or its affiliates.

25.56 UPDATE_REPLICATION_SERVER 54
25.57 UPDATE_SBT_ATTRIBUTE_SET 55
25.58 UPDATE_SBT_JOB_TEMPLATE 56
25.59 UPDATE_SBT_LIBRARY 57
25.60 UPDATE_STORAGE_LOCATION 58

26 Recovery Appliance View Reference

26.1 RA_ACTIVE_SESSION 2
26.2 RA_API_HISTORY 3
26.3 RA_CONFIG 4
26.4 RA_DATABASE 4
26.5 RA_DATABASE_HISTORY 8
26.6 RA_DATABASE_SYNONYM 9
26.7 RA_DATABASE_STORAGE_USAGE 10
26.8 RA_DB_ACCESS 10
26.9 RA_DISK_RESTORE_RANGE 11
26.10 RA_EM_SBT_JOB_TEMPLATE 12
26.11 RA_ENCRYPTION_INFO 13
26.12 RA_INCIDENT_LOG 13
26.13 RA_INCOMING_BACKUP_PIECES 14
26.14 RA_POLLING_FILES 14
26.15 RA_POLLING_POLICY 15
26.16 RA_PROTECTION_POLICY 16
26.17 RA_PURGING_QUEUE 17
26.18 RA_RECOVERY_COMPLIANCE 18
26.19 RA_REPLICATION_CONFIG 18
26.20 RA_REPLICATION_DATABASE 19
26.21 RA_REPLICATION_PAIR 20
26.22 RA_REPLICATION_POLICY 21
26.23 RA_RESTORE_RANGE 22
26.24 RA_REQUEST_BACKUP 23
26.25 RA_SBT_ATTRIBUTE_SET 23
26.26 RA_SBT_JOB 24
26.27 RA_SBT_LIBRARY 25
26.28 RA_SBT_MIRROR 26
26.29 RA_SBT_RESTORE_RANGE 26
26.30 RA_SBT_TASK 27
26.31 RA_SBT_TEMPLATE_MDF 28
26.32 RA_SERVER 29
26.33 RA_STORAGE_HISTOGRAM 29
26.34 RA_STORAGE_LOCATION 30

Copyright © 2014, 2026, Oracle and/or its affiliates.

26.35 RA_STORAGE_LOCATION_HISTORY 30
26.36 RA_TASK 31
26.37 RA_TIMER_TASK 33
26.38 RA_TIME_USAGE 33

27 rastat Utility Reference

27.1 rastat Command Syntax 1
27.2 Options 1

28 Recovery Appliance Error Message Reference

28.1 Recovery Appliance Error Message Reference (Continued) 16

Glossary

Index

Copyright © 2014, 2026, Oracle and/or its affiliates.

List of Figures

1-1	Full and Incremental Backups to Tape	2
1-2	RECOVER COPY on Disk, and Backup to Tape	2
1-3	Third-Party Deduplicating Appliance	3
1-4	Third-Party Copy-on-Write Snapshot	4
1-5	Recovery Appliance Environment	6
1-6	One-Way Replication	9
1-7	Backups to Tape Without Using Recovery Appliance	10
1-8	Backups to Tape Using Recovery Appliance	11
1-9	Delta Push and Delta Store	14
1-10	Recovery Appliance with Oracle Data Guard	19
2-1	Sample Recovery Appliance Environment	2
2-2	RASYS and Recovery Appliance User Accounts	5
2-3	Recovery Appliance Backup Modules	9
2-4	Redo Log Transmission	14
2-5	Recovery Appliance Metadata Database	15
2-6	Delta Pools in Delta Store	16
2-7	DELTA Storage Location	19
2-8	Backup Polling	21
2-9	Data Encryption Techniques	35
4-1	Storage Locations Page	5
7-1	Protection Policies	2
7-2	Protection Policies Page	5
7-3	Protection Policy Tasks in Recovery Appliance Workflow	6
7-4	Create Protection Policy Page	10
8-1	Protected Database Access	2
8-2	Database Access Configuration Tasks in the Recovery Appliance Workflow	2
9-1	Media Managers Page	15
9-2	Media Managers Page	19
9-3	Edit Media Manager Library Screen	20
9-4	Recovery Appliance Create Copy-to-Media Job Template Page	24
9-5	Tape Backup Jobs Example	27
11-1	Flow for Backups to Cloud Storage	3
15-1	Simple Replication Topology	1
15-2	Databases Replicating to One Recovery Appliance	6
15-3	Databases Replicated to Multiple Recovery Appliances	7
15-4	Different Protection Policies on Each Recovery Appliance	8

Copyright © 2014, 2026, Oracle and/or its affiliates.

15-5	Cascaded Replication, with Different Protection Policies on Each Recovery Appliance	9
15-6	Replication Page	23
15-7	Create Protection Policy Page	25
15-8	Protection Policy Advanced Parameters	26
15-9	Create Replication Server Page	27
15-10	Overview of Manual Configuration for Replication	29
16-1	Backups Replicated to two Recovery Appliances	24
16-2	Backup Anywhere Mode for Data Guard	24
16-3	Replication Request Mode of Backup Anywhere	25
16-4	Read-Only Mode of Backup Anywhere	26
17-1	Monitoring Tasks in the Recovery Appliance Workflow	4
18-1	Oracle Analytics Publisher Home Screen	1
18-2	Oracle Analytics Catalog	4
18-3	Reporting Tasks in the Recovery Appliance Workflow	6
24-1	Migration between two Recovery Appliances	5
24-2	Rack Migration with Bi-Directional Replication	6

Copyright © 2014, 2026, Oracle and/or its affiliates.

List of Tables

2-1	User Accounts in the Recovery Appliance Environment	4
2-2	Default Protection Policies	9
2-3	Protection Policy Attributes (subset)	10
2-4	Support for Incremental Forever with RMAN Encryption and RMAN Compression	36
7-1	DBMS_RA Protection Policy Procedures	5
7-2	Recovery Catalog Views for Protection Policies	6
7-3	Protection Policy Attributes (subset)	7
8-1	DBMS_RA Protected Database Access Procedures	10
8-2	Recovery Catalog Views for Protected Database Access	10
9-1	Recovery Appliance Objects for Copying Backups to Tape	5
9-2	DBMS_RA Procedures Associated with Tape/Cloud/Archive Backup Operations	9
9-3	Values for the STATUS Column of RA_SBT_LIBRARY	34
10-1	Prerequisites for Tape to Cloud Conversion	2
13-1	DBMS_RA Procedures Associated with Tape/Cloud/Archive Backup Operations	4
15-1	Principal RACLI Commands for Replication	13
15-2	Principal Procedures Relevant for Replication	28
15-3	Views for Replication	54
24-1	Principal RACLI Commands for Migration	3
25-1	DBMS_RS Package Subprograms	1
25-2	ABORT Parameters	5
25-3	ABORT_RECOVERY_APPLIANCE Parameters	6
25-4	ADD_DB Parameters	7
25-5	ADD_REPLICATION_SERVER Parameters	7
25-6	CONFIG Parameters	9
25-7	COPY_BACKUP Parameters	10
25-8	COPY_BACKUP_PIECE Parameters	12
25-9	CREATE_ARCHIVAL_BACKUP Parameters	14
25-10	CREATE_POLLING_POLICY Parameters	16
25-11	CREATE_PROTECTION_POLICY Parameters	17
25-12	CREATE_ARCHIVAL_BACKUP Parameters	21
25-13	CREATE_REPLICATION_SERVER Parameters	22
25-14	CREATE_SBT_ATTRIBUTE_SET Parameters	24
25-15	CREATE_SBT_JOB_TEMPLATE Parameters	25
25-16	CREATE_SBT_JOB_TEMPLATE Parameters	27
25-17	CREATE_SBT_LIBRARY Parameters	28
25-18	DELETE_DB Parameters	29

Copyright © 2014, 2026, Oracle and/or its affiliates.

25-19 DELETE_POLLING_POLICY Parameters 29
25-20 DELETE_PROTECTION_POLICY Parameters 30
25-21 DELETE_REPLICATION_SERVER Parameters 30
25-22 DELETE_SBT_ATTRIBUTE_SET Parameters 31
25-23 DELETE_SBT_JOB_TEMPLATE Parameters 31
25-24 DELETE_SBT_LIBRARY Parameters 32
25-25 ESTIMATE_SPACE Parameters 32
25-26 GET_REDO_TRANSPORT_LAG Parameters 33
25-27 GRANT_DB_ACCESS Parameters 33
25-28 KEY_REKEY Parameters 34
25-29 KEY_REKEY Parameters 34
25-30 KEY_REKEY Parameters 35
25-31 MIGRATE_TAPE_BACKUP Parameters 35
25-32 MOVE_BACKUP Parameters 36
25-33 MOVE_BACKUP_PIECE Parameters 37
25-34 PAUSE_REPLICATION_DATABASE Parameters 39
25-35 PAUSE_REPLICATION_SERVER Parameters 40
25-36 PAUSE_SBT_LIBRARY Parameters 40
25-37 POPULATE_BACKUP_PIECE Parameters 41
25-38 QUEUE_SBT_BACKUP_TASK Parameters 41
25-39 REMOVE_REPLICATION_SERVER Parameters 42
25-40 RENAME_DB Parameters 42
25-41 RESET_ERROR Parameters 43
25-42 RESUME_DB Parameters 44
25-43 RESUME_REPLICATION_DATABASE Parameters 44
25-44 RESUME_REPLICATION_SERVER Parameters 45
25-45 RESUME_SBT_LIBRARY Parameters 45
25-46 REVOKE_DB_ACCESS Parameters 46
25-47 SET_SYSTEM_DESCRIPTION Parameters 46
25-48 SHUTDOWN Parameters 46
25-49 SHUTDOWN_RECOVERY_APPLIANCE Parameters 47
25-50 STARTUP Parameters 47
25-51 STARTUP_RECOVERY_APPLIANCE Parameters 48
25-52 SUSPEND_DB Parameters 49
25-53 UPDATE_ARCHIVAL_BACKUP_KEEP Parameters 49
25-54 UPDATE_DB Parameters 50
25-55 UPDATE_POLLING_POLICY Parameters 51

Copyright © 2014, 2026, Oracle and/or its affiliates.

25-56 UPDATE_PROTECTION_POLICY Parameters 52
25-57 UPDATE_REPLICATION_SERVER Parameters 55
25-58 UPDATE_SBT_ATTRIBUTE_SET Parameters 56
25-59 UPDATE_SBT_JOB_TEMPLATE Parameters 56
25-60 UPDATE_SBT_LIBRARY Parameters 58
25-61 UPDATE_STORAGE_LOCATION Parameters 59
26-1 Recovery Appliance Views 1
27-1 rastat Options 1

Copyright © 2014, 2026, Oracle and/or its affiliates.

Preface

Welcome to *Zero Data Loss Recovery Appliance Administrator's Guide*.

Audience

This guide is intended for customers and those responsible for data center site planning, configuration, and maintenance of Zero Data Loss Recovery Appliance, commonly known as Recovery Appliance.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customer access to and use of Oracle support services will be pursuant to the terms and conditions specified in their Oracle order for the applicable services.

Related Documents

For more information, see the following documents:

- *Zero Data Loss Recovery Appliance Owner's Guide*
- *Zero Data Loss Recovery Appliance Protected Database Configuration Guide*
- *Oracle Database Backup and Recovery User's Guide*
- *Oracle Secure Backup Administrator's Guide*

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.
$ prompt	The dollar sign ($) prompt indicates a command run as the `oracle` user.
# prompt	The pound (#) prompt indicates a command that is run as the `root` user.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Changes in *Zero Data Loss Recovery Appliance Administrator's Guide* Release 23.1

The following are features in RA 23.1.

- Convert existing Oracle Secure Backup tape configurations to equivalent cloud configurations using the `racli convert tape_to_cloud` command. See Converting a Tape Configuration to Cloud for details.
- Space-efficient, encrypted backups
  - Compressed and encrypted backups, while supporting the incremental forever strategy. For more information, see Configuring Space-Efficient, Encrypted Backups and Configuring Space-Efficient, Encrypted Backups
  - Auto-tune reserved space for compliance backups
  - Standby database registration to Recovery Appliance Catalog
- Replication and Migration
  - Database-level backup migration across Recovery Appliances
  - Archived backups synchronized across replication pairs
  - Replication setup and configuration automation
  - Configuring Virtual Air Gap. Establishes Recovery Appliance replication with a time-controlled limited network access, such as a data vault scenario.
  - Migrating Backups with RACLI. Simplifies the migration from a source Recovery Appliance and a destination Recovery Appliance. All backups on the source are migrated to the destination Recovery Appliance, preserving the backup history.
- Updating the Recovery Appliance Software
  - zdlra update has been improved with new RACLI operations.
  - `racli alter appliance` runs *out-of-place* patching steps that do not require the appliance to be offline.
  - `racli update appliance` calls `racli alter appliance` and remembers where it was interrupted in the upgrade process so that it can be resumed at that point when called again.

The following are changes in *Zero Data Loss Recovery Appliance Administrator's Guide* for ZDLRA release 23.1.

- TLS Overview and Configuration chapter has improved instructions on various ways to create the trusted and signed certificates and to enable TLS network encryption.
- Replicating Backups with Recovery Appliance chapter has new instructions for using RACLI for replication management, and for configuring TLS and replication.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Migrating Backups with RACLI chapter is new. For decommissioning purposes, this chapterr describes how to migrate all of the backup data from an one Recovery Appliance to different Recovery Appliance.

- Oracle Enterprise Manager Cloud Control supports:.
  - Managing Recovery Window Compliance
  - Managing Compliance Holds: This software release addresses US government regulation SEC 17a-4(f) regarding recovery window compliance and *"legal holds"*. It provides enhanced APIs and protection policy attributes to allow this.
  - Archival Backups: Long-term archival backups: the ability for Recovery Appliance to send a backup of a single point in time, SCN, or tag to tertiary storage such as tape or cloud.
- Accessing Recovery Appliance Reports : This describes how to access the pre-created Oracle Business Intelligence (BI) Publisher reports specific to the Recovery Appliance that now reside in Oracle Analytics Publisher and are consolidated with other analytic reports.
- New API commands are:
  - CREATE_ARCHIVAL_BACKUP
  - CREATE_PROTECTION_POLICY
  - CREATE_RESTORE_POINT
  - GET_REDO_TRANSPORT_LAG
  - PAUSE_REPLICATION_DATABASE
  - PAUSE_REPLICATION_SERVER
  - RESUME_REPLICATION_DATABASE
  - RESUME_REPLICATION_SERVER
  - SUSPEND_DB
  - UPDATE_ARCHIVAL_BACKUP_KEEP
- New views are:
  - RA_DATABASE
  - RA_DATABASE_HISTORY
  - RA_RECOVERY_COMPLIANCE
  - RA_REPLICATION_CONFIG
  - RA_REPLICATION_DATABASE
  - RA_REPLICATION_PAIR
  - RA_REPLICATION_POLICY
  - RA_STORAGE_LOCATION_HISTORY
- Updated views are:
  - RA_API_HISTORY
  - RA_CONFIG
  - RA_DB_ACCESS
  - RA_ENCRYPTION_INFO
  - RA_INCIDENT_LOG

Copyright © 2014, 2026, Oracle and/or its affiliates.

- RA_INCOMING_BACKUP_PIECES
- RA_RESTORE_RANGE
- RA_SBT_LIBRARY
- RA_SERVER
- RA_STORAGE_LOCATION
- RA_TASK

Copyright © 2014, 2026, Oracle and/or its affiliates.

1

Introduction to Recovery Appliance

The cloud-scale Zero Data Loss Recovery Appliance, commonly known as Recovery Appliance, is an Engineered System designed to dramatically reduce data loss and backup overhead for all Oracle databases in the enterprise. Integrated with Recovery Manager (RMAN), the Recovery Appliance enables a centralized, incremental-forever backup strategy for large numbers of databases, using cloud-scale, fault-tolerant hardware and storage. The Recovery Appliance continuously validates backups for recoverability.

This chapter contains the following topics:

- Traditional Database Backup Techniques
- Data Protection Challenges in the Modern Enterprise
- Oracle's Recovery Appliance Solution
- What's Next?

1.1 Traditional Database Backup Techniques

All production Oracle databases require data protection. Oracle provides RMAN as its preferred backup solution. Most enterprises have adopted one or more of the database backup strategies described in this section:

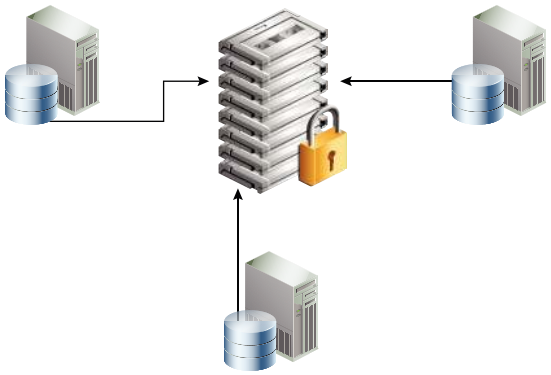
- Weekly Full and Daily Incremental Backups
- Incremental Backups and RECOVER COPY
- Full Backups to a Third-Party Deduplicating Appliance
- Third-Party Storage Snapshots

1.1.1 Weekly Full and Daily Incremental Backups

One popular approach, shown in Figure 1-1, is to use RMAN to take a weekly full backup, and then daily incremental backups. To improve incremental backup performance, Oracle recommends enabling block change tracking. These backups occur when activity on the database is lowest.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Figure 1-1 Full and Incremental Backups to Tape



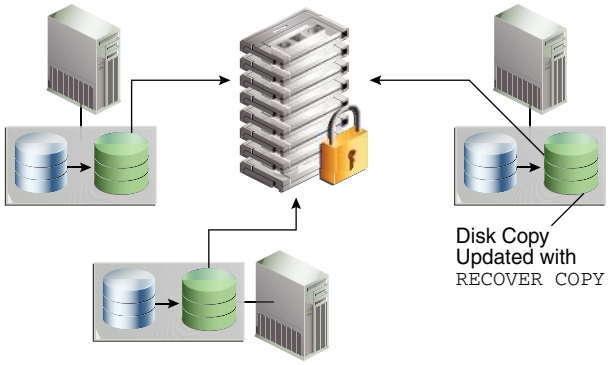
An advantage of this technique is that backup windows, which affect the production server, are relatively brief on the days when incremental backups occur. A disadvantage is that when the database is continuously active, as when serving multiple global time zones, no easily accommodating backup window is available.

One solution is to set up Oracle Data Guard, and then back up the standby database, thereby removing the backup load from the production server. However, protecting all databases with Oracle Data Guard is often impractical.

1.1.2 Incremental Backups and RECOVER COPY

The RMAN technique shown in Figure 1-2 makes daily incremental backups, and then uses the `RECOVER COPY` command to merge the incremental changes into the full database copy. In this way, the database copy on disk is "rolled forward" every day.

Figure 1-2 RECOVER COPY on Disk, and Backup to Tape



This technique has the following advantages:

- Only one initial full backup is required, which reduces the total weekly backup window time.
- An RMAN `SWITCH` command can point the control file to the database copy, which turns the copy into an actual database file, and thus eliminates the `RESTORE` step.

Some disadvantages are as follows:

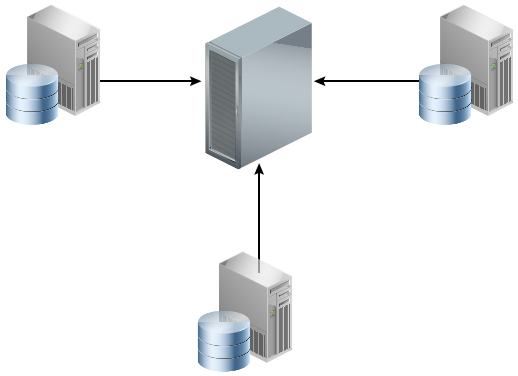
Copyright © 2014, 2026, Oracle and/or its affiliates.

- You must have sufficient disk space to keep a copy of the whole database on disk, and the archived redo log files required to recover it.
- Only one physical copy of the database exists. You select the point in time at which to keep the copy, so you can recover to subsequent points in time. For example, to restore to any point in time within the past week, your physical copy must be older than `SYSDATE-7`. The disadvantages are:
  - You cannot recover to a time earlier than the time at which you maintain the database copy.
  - The closer your recovery point in time is to the current time, the more incremental backups you must restore and apply to the copy. This technique adds time to the overall recovery time objective.
- The database copy cannot be compressed or encrypted.

1.1.3 Full Backups to a Third-Party Deduplicating Appliance

As an alternative to RMAN incremental backups and tape drives, some customers use third-party deduplicating appliances to process backup streams. Figure 1-3 depicts three databases writing to a centralized third-party appliance.

Figure 1-3 Third-Party Deduplicating Appliance



This technique has the following advantages:

- A central backup location serves all databases in the environment.
- The third-party software searches for patterns at the byte and sub-byte level to eliminate redundant data from backup to backup. For example, if a full database backup is almost identical to the backup taken a week before, then the software can attempt to prune the redundant bits from the incoming backup stream.
- To reduce network load, one optional technique utilizes source-side deduplication so that backup streams are deduplicated on the database host instead of the third-party appliance. Typically, this technique relies on an RMAN SBT plug-in.

Some disadvantages are as follows:

- These third-party appliances do not recognize or validate Oracle AI Database blocks. From the perspective of the appliance, a database backup is the same as a file system backup: a stream of bytes.

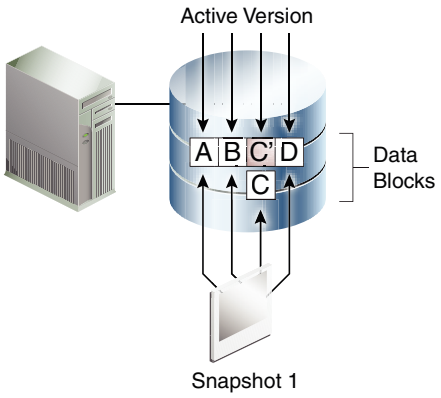
Copyright © 2014, 2026, Oracle and/or its affiliates.

- Deduplication is only effective for full database backups that have a high degree of redundancy. Strategies that use incremental backups often do not achieve good deduplication ratios.
- The third-party appliance dictates which Oracle AI Database features to use rather than the other way around. Often, adapting to the requirements of the appliance means rewriting existing backup scripts.

1.1.4 Third-Party Storage Snapshots

A third-party storage snapshot is a set of pointers to storage blocks (*not* Oracle blocks) that existed when the snapshot was created. The virtual copies reside on the same storage array as the original data. Figure 1-4 depicts a copy-on-write snapshot, which is a type of third-party snapshot. After a snapshot is taken, when the first change to a storage block occurs, the array copies the before-image block to a new location on disk (`C`) and writes the new block (`C'`) to the original location.

Figure 1-4 Third-Party Copy-on-Write Snapshot



This technique has the following advantages:

- An initial copy of the database is not necessary because snapshots are not stored as physical copies of blocks. Thus, less storage is consumed than in RMAN strategies.
- Snapshots can be extremely fast. You put the database in backup mode (unless storage does *not* meet the requirements for snapshot storage optimization), and then take the snapshot. The snapshot needs to store physical blocks only when the blocks change, so a backup of an unchanged file is a metadata-only operation.
- Snapshots use storage efficiently. A backup of a file with a single changed block requires only one additional version of the block to be stored—either the old version or new version of the block, depending on the snapshot technique.

Some disadvantages are as follows:

- Snapshots have no knowledge of an Oracle AI Database block structure, and thus cannot validate Oracle blocks.
- Because snapshots reside on the same storage array as the source database, they are vulnerable to storage failures and data corruptions. If the array is inaccessible, or if the storage contains data block corruptions, then the snapshots cannot be used for recovery.
- Restoring a snapshot in place voids all snapshots that were taken after it unless the snapshot is fully restored to an alternate location.

Copyright © 2014, 2026, Oracle and/or its affiliates.

> **See Also**
>
> *Oracle Database Backup and Recovery User's Guide* to learn more about using Storage Snapshot Optimization to take third-party snapshots of the database

1.2 Data Protection Challenges in the Modern Enterprise

The role of information technology in the modern business is going through a tremendous transformation. The key drivers for this transformation are:

- Data growth

  Many organizations continue to experience exponential growth, which creates a greater challenge for efficient data management and protection. What works well for dozens of databases may not work well for hundreds or thousands of databases, often running on different platforms and on multiple physical servers.

- Real-time analytics

  Organizations are increasingly dependent on data analysis for critical real-time decisions. This dependency increases the pressure to maintain data integrity and prevent data loss.

- Continuous global availability

  Many databases provide 24/7 access across multiple time zones, which means that databases are continuously active.

The protection strategies described in "Traditional Database Backup Techniques" are not designed to solve the challenges created by this transformation. Enterprises find themselves without a consistent backup and recovery strategy. The following shortcomings are common to most or all of the traditional backup techniques:

- Data loss exposure

  A database is only recoverable to its last valid backup, which may have occurred hours or days ago. In addition, storage snapshots and third-party appliances cannot validate Oracle data blocks, and so cannot detect Oracle block-level corruptions.

- Long backup windows

  As database sizes increase, the lengths of the backup windows also increase, creating additional load on production systems. Critical databases cannot afford to be deprived of resources used for daily backups and related maintenance activities.

- Lack of backup validation

  Because most third-party backup snapshot and Recovery Appliances lack Oracle integrated data block and database backup validation, restore and recovery operations tend to fail. Such failures result in extended downtime and potentially larger data loss.

- Lack of end-to-end visibility

  As the number of databases increases exponentially, so the ease of manageability decreases. Backup scripts proliferate and change. New DBAs may struggle to understand what the legacy scripts do. Questions about the status, backup location, and recovery point objective (RPO) of a particular database become harder to answer.

The traditional techniques fail to provide a comprehensive and efficient Oracle-integrated data protection solution that meets the demands of a large-scale, enterprise Oracle environment. A new approach is required.

Copyright © 2014, 2026, Oracle and/or its affiliates.

1.3 Oracle's Recovery Appliance Solution

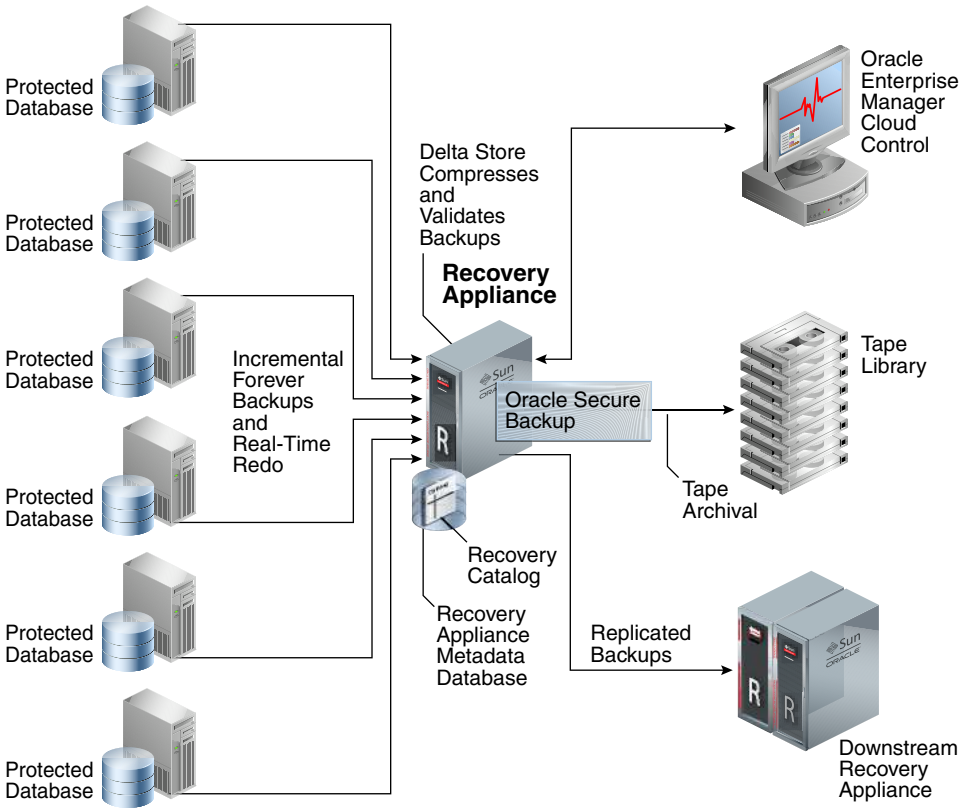
Recovery Appliance is a cloud-scale Engineered System designed to protect all Oracle databases across the enterprise. Most database backup and restore processing is performed by the centralized Recovery Appliance, making storage utilization, performance, and manageability of backups more efficient.

The Recovery Appliance stores and manages backups of multiple Oracle databases in a unified disk pool, using an RMAN incremental-forever strategy. The Recovery Appliance continually compresses, deduplicates, and validates backups at the database block level, while creating virtual full backup on demand.

A virtual full backup is a complete database image as of one distinct point in time, maintained efficiently through Recovery Appliance indexing of incremental backups from protected databases. A virtual full backup can correspond to any incremental backup that was received.

Figure 1-5 shows an overview of a sample Recovery Appliance environment.

Figure 1-5 Recovery Appliance Environment



As shown in Figure 1-5, a protected database is a client database that backs up data to a Recovery Appliance. Each protected database uses the Zero Data Loss Recovery Appliance Backup Module (Recovery Appliance Backup Module) for its backups. This module is an Oracle-supplied SBT library that RMAN uses to transfer backup data over the network to the Recovery Appliance.

The Recovery Appliance metadata database, which resides on each Recovery Appliance, manages metadata stored in the RMAN recovery catalog, and backups located in the

Copyright © 2014, 2026, Oracle and/or its affiliates.

Recovery Appliance storage location. The catalog is required to be used by all protected databases that send backups to Recovery Appliance.

> **Note**
>
> Databases may use Recovery Appliance as their recovery catalog without also using it as a backup repository.

Administrators use Oracle Enterprise Manager Cloud Control (Cloud Control) to manage and monitor the environment. Cloud Control provides a "single pane of glass" view of the entire backup lifecycle for each database, whether backups reside on disk, tape, or another Recovery Appliance.

Recovery Appliance provides the following benefits:

- Elimination of Data Loss
- Minimal Backup Overhead
- Improved End-to-End Data Protection Visibility
- Cloud-Scale Protection

> **See Also**
>
> "The Recovery Appliance Environment"

> **Note**
>
> Oracle Secure Backup (OSB) 19.1 desupport was announced on May 1, 2026 with premier support end date on September 30, 2027. See My Oracle Support note PNEWS3035 for more information. Please refer to Lifetime Support Policy: Oracle Technology Products for updated support timeframes.
>
> Oracle recommends that database backups to cloud take advantage of Oracle Database Zero Data Loss Cloud Protect, which uses Oracle Zero Data Loss Autonomous Recovery Service in OCI. Cloud Protect offers an efficient incremental forever backup strategy, real-time transaction protection, logically air-gapped immutable backups, and fast, point-in-time recovery.
>
> For customers who require database backups to Amazon S3, Oracle Database Cloud Backup for Amazon S3 may be used. This offering is similar to Oracle Secure Backup Cloud Module in that it is fully integrated with Recovery Manager (RMAN) and enables you to back up your Oracle database to Amazon S3. You can exchange your Oracle Secure Backup license for an Oracle Database Cloud Backup for Amazon S3 license on a 1:1 basis. See Oracle Database Cloud Backup for Amazon S3 for more information.
>
> As with Oracle Secure Backup Cloud Module, Oracle Database Cloud Backup for Amazon S3 can be used free of charge on Oracle Zero Data Loss Recovery Appliance (ZDLRA) for the sole purpose of backing up data to and restoring data from S3-compatible storage devices that are in the same campus as ZDLRA.

Copyright © 2014, 2026, Oracle and/or its affiliates.

1.3.1 Elimination of Data Loss

The Recovery Appliance uses various mechanisms to protect against different types of data loss, including physical block corruption. This section contains the following topics:

- Protection of Ongoing Transactions
- Secure Replication
- Autonomous Tape Archival
- End-to-End Data Validation

1.3.1.1 Protection of Ongoing Transactions

In traditional backup approaches, if the online redo log is lost, then media recovery loses all changes after the most recent available archived redo log file or incremental backup. A recovery point objective (RPO) of a day or more that might result from a traditional approach may be unacceptable.

Recovery Appliance solves the RPO problem through a continuous transfer of redo changes to the appliance from a protected database. This operation is known as real-time redo transport. Using delta push, the Recovery Appliance is a remote destination for asynchronous redo transport services from Oracle Database 11*g* and Oracle Database 12*c* databases.

> **Note**
>
> This technology is based on the real-time redo transport algorithms of Oracle Data Guard. To avoid degrading the performance of the protected database, protected databases transfer redo asynchronously to the Recovery Appliance. If a protected database is lost, zero to subsecond data loss is expected in most cases.

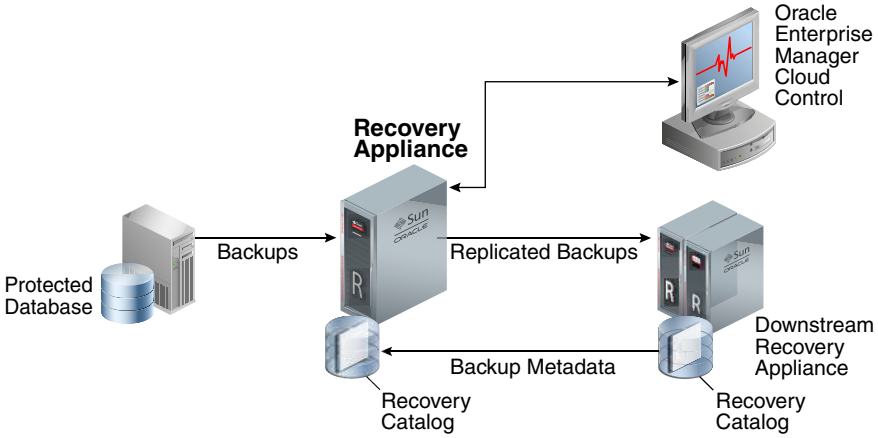
> **See Also**
>
> - "Real-Time Redo Transport" to learn more about real-time redo transport
> - "Delta Push"
> - *Oracle Data Guard Concepts and Administration* for information about Oracle Data Guard redo transport

1.3.1.2 Secure Replication

To protect against server or site outage, one Recovery Appliance can replicate backups to a different Recovery Appliance. Figure 1-6 shows the simplest form of replication, called one-way Recovery Appliance replication, in which an upstream Recovery Appliance (backup sender) transfers backups to a downstream Recovery Appliance (backup receiver).

Copyright © 2014, 2026, Oracle and/or its affiliates.

Figure 1-6 One-Way Replication



In Figure 1-6, a protected database sends an incremental backup to the Recovery Appliance, which then queues it for replicating to the downstream Recovery Appliance. When the upstream Recovery Appliance sends the incremental backup to the downstream Recovery Appliance, it creates a virtual full backup as normal. The downstream Recovery Appliance creates backup records in its recovery catalog. When the upstream Recovery Appliance requests the records, the downstream Recovery Appliance propagates the records back.

If the local Recovery Appliance cannot satisfy virtual full backup requests, then it automatically forwards them to the downstream Recovery Appliance, which sends virtual full backups to the protected database. DBAs use RMAN as normal, without needing to understand where or how the backup sets are stored.

> **See Also**
>
> "Recovery Appliance Replication"

1.3.1.3 Autonomous Tape Archival

A robust backup strategy protects data against intentional attacks, unintentional user errors (such as file deletions), and software or hardware malfunctions. Tape libraries provide effective protection against these possibilities.

Figure 1-7 show the traditional technique for tape backups, with a media manager installed on each host.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Figure 1-7 Backups to Tape Without Using Recovery Appliance

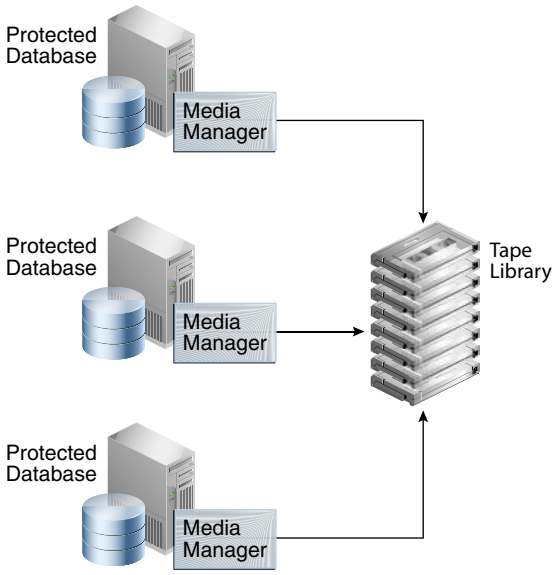
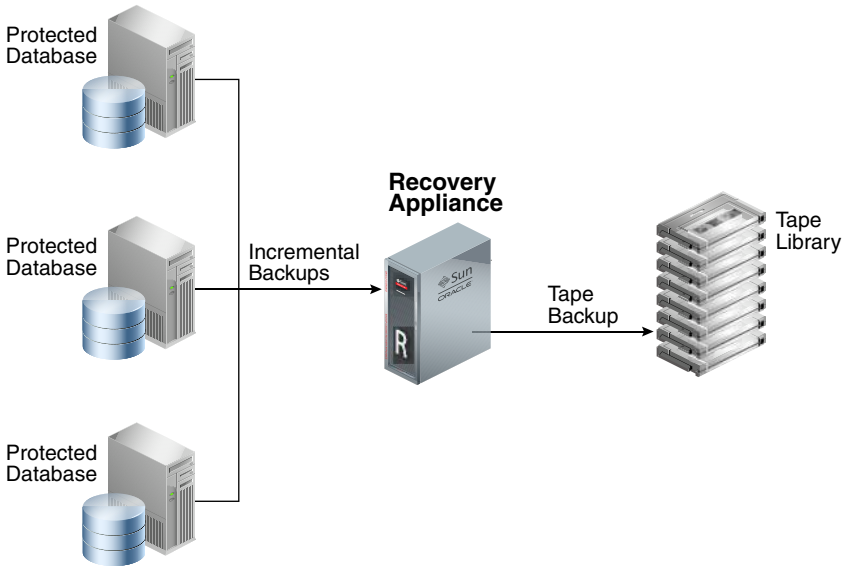


Figure 1-8 shows the Recovery Appliance technique for tape backups. The fundamental difference in the two approaches is that the Recovery Appliance backs up to tape, *not* the protected databases. The Recovery Appliance comes with preinstalled Oracle Secure Backup software, and supports optional Fibre Channel cards. Thus, installation of a media manager is not necessary on the protected database hosts.

> **Note**
>
> Oracle Secure Backup (OSB) 19.1 desupport was announced on May 1, 2026 with premier support end date on September 30, 2027. See My Oracle Support note PNEWS3035 for more information. Please refer to Lifetime Support Policy: Oracle Technology Products for updated support timeframes.
>
> Oracle recommends that database backups to cloud take advantage of Oracle Database Zero Data Loss Cloud Protect, which uses Oracle Zero Data Loss Autonomous Recovery Service in OCI. Cloud Protect offers an efficient incremental forever backup strategy, real-time transaction protection, logically air-gapped immutable backups, and fast, point-in-time recovery.
>
> For customers who require database backups to Amazon S3, Oracle Database Cloud Backup for Amazon S3 may be used. This offering is similar to Oracle Secure Backup Cloud Module in that it is fully integrated with Recovery Manager (RMAN) and enables you to back up your Oracle database to Amazon S3. You can exchange your Oracle Secure Backup license for an Oracle Database Cloud Backup for Amazon S3 license on a 1:1 basis. See Oracle Database Cloud Backup for Amazon S3 for more information.
>
> As with Oracle Secure Backup Cloud Module, Oracle Database Cloud Backup for Amazon S3 can be used free of charge on Oracle Zero Data Loss Recovery Appliance (ZDLRA) for the sole purpose of backing up data to and restoring data from S3-compatible storage devices that are in the same campus as ZDLRA.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Figure 1-8 Backups to Tape Using Recovery Appliance



When Recovery Appliance executes a copy-to-tape job for a virtual full backup, it constructs the physical backup sets, and copies them to tape, and then writes the metadata to the recovery catalog. If desired, the Recovery Appliance can also copy successive incremental backups and archived redo log file backups to tape. Whereas the backup on the Recovery Appliance is virtual, the backup on tape is a non-virtual, full physical backup. The Recovery Appliance automatically handles requests to restore backups from tape, with no need for administrator intervention.

The advantages of the Recovery Appliance tape solution are as follows:

- The Recovery Appliance performs all tape copy operations automatically, with no performance load on the protected database host.
- Tape backups are optimized. Recovery Appliance intelligently gathers the necessary blocks to create a non-virtual, full backup for tape.
- Oracle Secure Backup is preinstalled, eliminating the need for costly third-party media managers.

> **Note**
>
> You may deploy tape backup agents from third-party vendors on the Recovery Appliance for integration with existing tape backup software and processes. In this configuration, the agents must connect to their specialized media servers, which must be deployed externally to the Recovery Appliance.

- Tape drives and tape libraries function more efficiently because Recovery Appliance is a single large centralized system with complete control over them. In other tape solutions, hundreds or thousands of databases can contend for tape resources in an uncoordinated manner.

Copyright © 2014, 2026, Oracle and/or its affiliates.

See Also

- Copying Backups to Tape with Recovery Appliance
- *Oracle Secure Backup Administrator's Guide*

1.3.1.4 End-to-End Data Validation

A basic principle of backup and recovery is to ensure that backups can be restored successfully. To ensure that there are no physical corruptions within the backed-up data blocks, backups require regular validation. Validation typically involves running an RMAN RESTORE VALIDATE job regularly, along with running periodic full restore and recovery operations to a separate machine.

Recovery Appliance provides end-to-end block validation, which occurs in the following stages of the workflow:

- Recovery Appliance validation

  The Recovery Appliance automatically validates the backup stream during the backup ingest phase, before writing the backups to disk. The Recovery Appliance also validates the backup before sending it back to the original or alternate database server during the restore phase. Therefore, no manual `RESTORE VALIDATE` step is required.

  In addition, a background task running on the Recovery Appliance periodically validates the integrity of the virtual full backups in the delta pools (see "Delta Pools"). The goal of this task is to check each block of each virtual full backup of each protected database and to work behind the scenes when minimal activity is occurring. By default, the validation task runs every 14 days following the last completed validation of a database's current set of backups on disk.

  Just as with data file backups, the Recovery Appliance validates the integrity of redo log blocks during every operation, including receiving redo from the protected database, and storing it in compressed archived log backup sets.

- Oracle Automatic Storage Management (ASM)

  Oracle ASM stores the backup and redo data for the Recovery Appliance. Oracle ASM mirrored copies provide redundancy (see "Recovery Appliance Storage Locations").

  If a corrupted block is read on the primary mirror, the Recovery Appliance automatically repairs the block from the mirrored copy. This mechanism resolves most isolated block corruption cases.

- Tape library

  Recovery Appliance validates blocks when it copies them to tape, and also when it restores them from tape (see "Tape Archival").

- Downstream Recovery Appliance in a replication configuration

  If you configure replication, then the downstream Recovery Appliance validates data during the backup ingest and restore phases (see "How a Downstream Recovery Appliance Processes Backups").

None of the preceding backup validation processes occur on the production database hosts, thus freeing production resources for more critical operational workloads.

Copyright © 2014, 2026, Oracle and/or its affiliates.

> **Note**
>
> Oracle Maximum Availability Architecture best practices recommend that you still perform periodic full database recovery tests to verify operational practices and to detect issues that might occur only during media recovery.

> **See Also**
>
> - "CONFIG" for information about the `validate_db_days` configuration parameter
> - "RA_DATABASE" for information about the `RA_DATABASE.LAST_VALIDATE` column

1.3.2 Minimal Backup Overhead

In traditional database backup techniques, the Oracle database host performs the brunt of the processing. Agents for disk backup, tape backup, and deduplication may all be running on the host. Furthermore, all backup operations—compression, validation, deletion, merging, and so on—occur on the database host. This overhead can greatly degrade database performance.

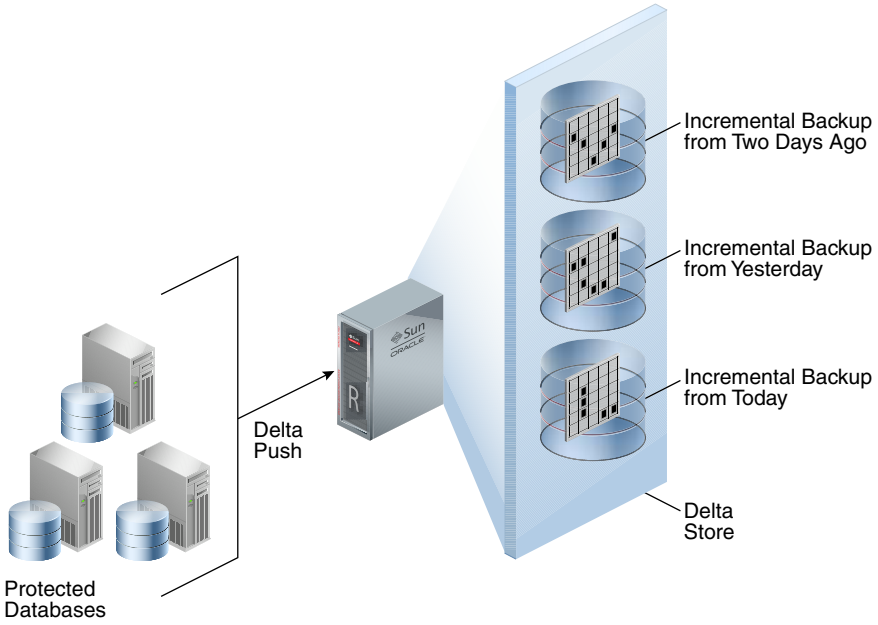
Recovery Appliance removes almost the entire load from the protected databases. The only backup operations required on the hosts, which could be primary database or standby database hosts, are sending incremental backups to the Recovery Appliance. The incremental-forever strategy reduces the backup window on the database hosts significantly. Recovery Appliance handles backup processing, tape operations, data integrity checks, and routine maintenance.

> **Note**
>
> Recovery Appliance only supports backups of Oracle databases, not file system data or non-Oracle databases.

Recovery Appliance optimizes management of database changes using delta push and delta store, shown in Figure 1-9. The net result of delta push and delta store is that the problem of lengthening backup windows is eliminated. The DBA performs only fast incremental backups, and lets the Recovery Appliance manage the backup blocks.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Figure 1-9 Delta Push and Delta Store



> **See Also**
>
> "Traditional Database Backup Techniques"

1.3.2.1 Delta Push

This solution consists of two operations that run on each protected database: the incremental-forever backup strategy, and real-time redo transport. Both operations involve protected databases pushing changes to the Recovery Appliance.

In an incremental-forever strategy, only one incremental level 0 backup to the Recovery Appliance is required in the lifetime of each protected file. The initial level 0 backup does not contain committed undo blocks or currently unused blocks.

> **Note**
>
> The elimination of committed undo and currently unused blocks is only supported for SBT full backups to the Recovery Appliance or Oracle Secure Backup. It is not available for SBT backups to other backup products.

In normal operation, the Recovery Appliance automatically performs the following steps for each incremental level 1 backup:

1. Receives a scheduled incremental level 1 backup from each protected database
2. Validates the incoming backup to protect against physical block corruptions
3. Compresses the backup using specialized block-level algorithms

Copyright © 2014, 2026, Oracle and/or its affiliates.

4. Writes the backup to a delta store in a Recovery Appliance storage location

The incremental-forever strategy greatly reduces the backup window and overhead because no full backups are ever required after the initial incremental level 0 backup. If the strategy includes real-time redo transport, then backup windows are further reduced because traditional archived log backups are not necessary. Also, Recovery Appliance takes on the burden of validation, deduplication, and compression.

> **Note**
>
> Blocks compressed using table or Hybrid Columnar Compression remain compressed in the RMAN backup and during the Recovery Appliance ingest phase.

> **See Also**
>
> "Elimination of Data Loss"

1.3.2.2 Delta Store

The delta store is the key processing engine for Recovery Appliance. A protected database sends only one incremental level 0 backup of each data file to the Recovery Appliance. Following the initial full backup, all backups are highly efficient cumulative incremental backups.

As Recovery Appliance receives incremental backups, it indexes them and stores them in delta pools. Each separate data file backed up to the Recovery Appliance has its own separate delta pool (set of backup blocks). Recovery Appliance automatically manages the delta pools so that it can provide many virtual full backups.

1.3.2.2.1 Creation of Virtual Full Backups

To create a virtual full backup, Recovery Appliance converts an incoming incremental level 1 backup into a virtual representation of an incremental level 0 backup. A virtual full backup appears as an incremental level 0 backup in the recovery catalog. From the user's perspective, a virtual full backup is indistinguishable from a non-virtual full backup. Using virtual backups, Recovery Appliance provides the protection of frequent level 0 backups with only the cost of frequent level 1 backups.

> **Note**
>
> Recovery Appliance provides storage services, but not virtual full backups, for RMAN-encrypted backups (see "Archival and Encrypted Backups"). These backups are stored in their original encrypted format. Recovery Appliance can store, archive, and retrieve them just as it can for unencrypted RMAN backup sets.

1.3.2.2.2 Rapid Recovery Using Virtual Full Backups

Recovery Appliance uses virtual full backups to provide rapid recovery to any point in time, regardless of the amount of data being recovered. The on-disk recovery strategy of Recovery

Copyright © 2014, 2026, Oracle and/or its affiliates.

Appliance has the advantage that RMAN can recover virtual full backups to any point in time *without* applying incremental backups.

When a database is protected by the Recovery Appliance, RMAN must only restore a single level 0 backup for the day of the RPO, and then recover up to the last second using redo log files sent using the real-time redo transport feature. For example, if the recovery window is 7 days, and if the RPO is 5 days ago, then RMAN can restore a single virtual full (level 0) backup that is current to 5 days ago, and then recover it using redo—not level 1 incremental backups.

See Also

- "Delta Store"
- *Zero Data Loss Recovery Appliance Protected Database Configuration Guide* to learn more about the incremental-forever backup strategy
- *Zero Data Loss Recovery Appliance Protected Database Configuration Guide* to learn more about recovery strategies
- *Oracle Database Backup and Recovery User's Guide* to learn more about incremental backups

1.3.3 Improved End-to-End Data Protection Visibility

In traditional database backup techniques, management of the database, media server, and tape drives are often separated. For example, a DBA group may manage the databases, while a separate backup administrator group manages the backups, and a storage group manages the disk and tape devices. The overall process lacks visibility, which makes it difficult to manage backups for thousands of databases, each with different recovery requirements.

Cloud Control provides a complete, end-to-end view into the backup lifecycle managed by the Recovery Appliance, from the time the RMAN backup is initiated on the database, to when it is stored on disk, tape, or replicated to a downstream Recovery Appliance. Recovery Appliance monitoring and administration are enabled through installation of the Enterprise Manager for Zero Data Loss Recovery Appliance plug-in (Recovery Appliance plug-in).

Using Cloud Control to manage a Recovery Appliance provides the following benefits:

- Standard metrics such as overall backup performance, and aggregate or per-database space consumption
- Immediate alerts about any backup or Recovery Appliance issues

  For example, Cloud Control may alert the administrator if no backup is available to meet the defined RPO, or if corrupt backups are discovered.
- Status reports, enabled by BI Publisher, are useful for capacity planning and to identify protected databases that are not meeting recovery window goals

  For example, Recovery Appliance administrators can receive reports on historical space and network usage to identify backup volume and throughput trends. These trends may necessitate adding storage servers to an existing rack or connecting additional racks.

Although Cloud Control is the recommended user interface for Recovery Appliance administration, Oracle supplies the `DBMS_RA` PL/SQL package as a command-line alternative. Most tasks in this manual provide both Cloud Control and `DBMS_RA` techniques. For command-line monitoring and reporting, you can query the Recovery Appliance catalog views.

Copyright © 2014, 2026, Oracle and/or its affiliates.

See Also

- Traditional Database Backup Techniques
- Getting Started with Cloud Control for Recovery Appliance
- DBMS_RA Package Reference
- Recovery Appliance View Reference

1.3.4 Cloud-Scale Protection

Recovery Appliance scales at a cloud level, supporting tens to hundreds to thousands of databases across a data center. Essentially, Recovery Appliance enables you to create a private data protection cloud within the enterprise. The following technology components within Recovery Appliance make this possible:

- Policy-Based Data Protection Management
- Database-Aware Space Management
- Scalable Architecture

1.3.4.1 Policy-Based Data Protection Management

Recovery Appliance simplifies management through the protection policy. Benefits include the following:

- A protection policy defines recovery window goals that are enforced for each database for backups to the Recovery Appliance or a tape device.

  Using protection policies, you can group databases by recovery service tier. For example, databases protected by the Platinum policy require backups to be kept for 45 days on the Recovery Appliance and 90 days on tape, which means that backups aged 45 days or less exist on disk *and* tape, but backups older than 45 days are only on tape. Databases protected by the Gold policy require 35 days on the local Recovery Appliance and 90 days on tape. Optionally, you can define a maximum retention time within each policy to limit the space consumed, and to comply with service level agreements dictating that backups cannot be maintained for longer than a specified period.
- Protection policies are means of grouping databases, improving manageability.

  For example, you can configure Recovery Appliance replication or copy-to-tape for a specific protection policy, which means that the configuration applies to all databases associated with this policy. If you add a database to the policy, then the database automatically inherits the configurations and scheduling of the policy.

See Also

- "Protection Policies"
- Managing Protection Policies with Recovery Appliance

Copyright © 2014, 2026, Oracle and/or its affiliates.

1.3.4.2 Database-Aware Space Management

Using protection policies, the Recovery Appliance manages backup storage space according to the recovery window goal for each protected database. This granular, database-oriented space management approach eliminates the need to manage space at the storage-volume level, as third-party appliances do.

If space is available, then the Recovery Appliance may retain backups older than the recovery window goal, effectively extending the point-in-time recovery period. When space pressure exists, the Recovery Appliance uses predefined thresholds to purge backups. The Recovery Appliance automatically provisions space so that the recovery window goal for each database is met.

The Oracle Zero Data Loss Recovery Appliance with Release 21.1 supports immutable backups, which are a requirement of various government regulations such as SEC 17(a)-4f and others from FINRA, CFTC, MiFID II, which specify rules for compliance retention and legal holds.

The *"Recovery Window Compliance"* is a range of time that the Recovery Appliance will ensure databases can be recovered from their backups. When defined in the protection policy, newly created backups of that policy are held on the Recovery Appliance for that period of time. Recovery window compliance is different and more restrictive than the recover window goal.

If compliance features are not enabled, the protection policy can make use of *"autotuned reserved space"*, which has the Recovery Appliance automatically define and update the storage space.

> **See Also**
>
> "How Recovery Appliance Manages Storage Space"

1.3.4.3 Scalable Architecture

The approaches in "Traditional Database Backup Techniques" are prone to performance bottlenecks and multiplying points of failure. As the number of databases increases, so does the number of media servers, disk arrays, tape devices, and third-party appliances, and thus so does the overall complexity. The "add more devices" approach is not scalable. In contrast, Recovery Appliance can scale to accommodate increases in backup traffic, storage usage, and the number of databases by adding compute and storage resources in a simple, modular fashion.

> **See Also**
>
> *Zero Data Loss Recovery Appliance Owner's Guide* for information about adding storage servers

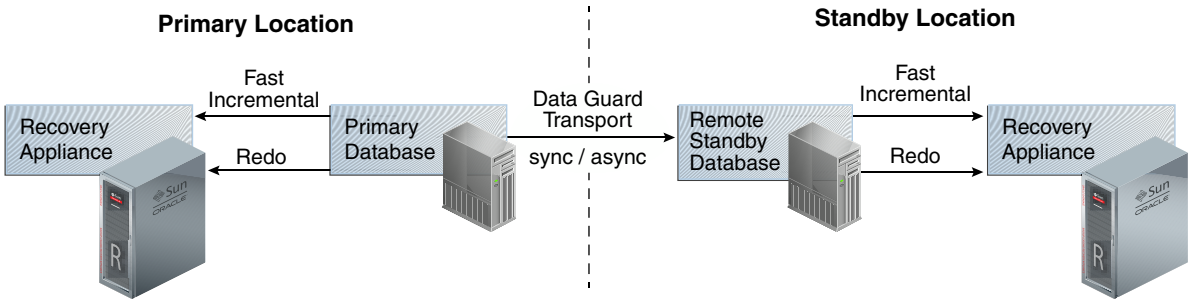
1.3.5 Maximum Availability: Recovery Appliance with Oracle Data Guard

Oracle Data Guard is a component of a high availability (HA) and disaster recovery solution that can be integrated with Recovery Appliance to provide maximum data protection. Oracle Data Guard minimizes service interruption and resulting data loss by maintaining a

Copyright © 2014, 2026, Oracle and/or its affiliates.

synchronized standby database for the protected database. When the primary system is unavailable, the standby immediately assumes the normal operations of the primary after a Data Guard failover operation, including backups to the local Recovery Appliance. Figure 1-10 shows an example of an environment with Recovery Appliance and Oracle Data Guard.

Figure 1-10 Recovery Appliance with Oracle Data Guard



In Figure 1-10, the primary and standby databases each send incremental backups to their local Recovery Appliance. The primary database sends real-time redo changes to both the local Recovery Appliance and the physical standby, and the standby cascades the redo changes to the remote Recovery Appliance. Each Recovery Appliance has backups and redo information for the same database, therefore either appliance can be used for RMAN restore and recovery operations.

See Also

- http://www.oracle.com/technetwork/database/availability/disaster-recovery-2526839.pdf to learn more about Recovery Appliance with Oracle Data Guard
- *Oracle Data Guard Concepts and Administration* for information about Oracle Data Guard

1.4 What's Next?

To begin using Recovery Appliance, refer to the following topics:

1. Optionally, read Recovery Appliance Architecture to obtain a more in-depth understanding of the principal components of the Recovery Appliance environment.
2. Read Recovery Appliance Workflow to learn about basic tools and tasks. Before you can use Recovery Appliance for data protection, you must perform the tasks described in the following topics:
   a. "Planning for Recovery Appliance"
   b. "Setup and Configuration for Recovery Appliance"
   c. "Maintenance Tasks for Recovery Appliance"

Copyright © 2014, 2026, Oracle and/or its affiliates.

2

Recovery Appliance Architecture

This chapter describes the basic architecture and concepts for Zero Data Loss Recovery Appliance, commonly known as Recovery Appliance.

This chapter contains the following topics:

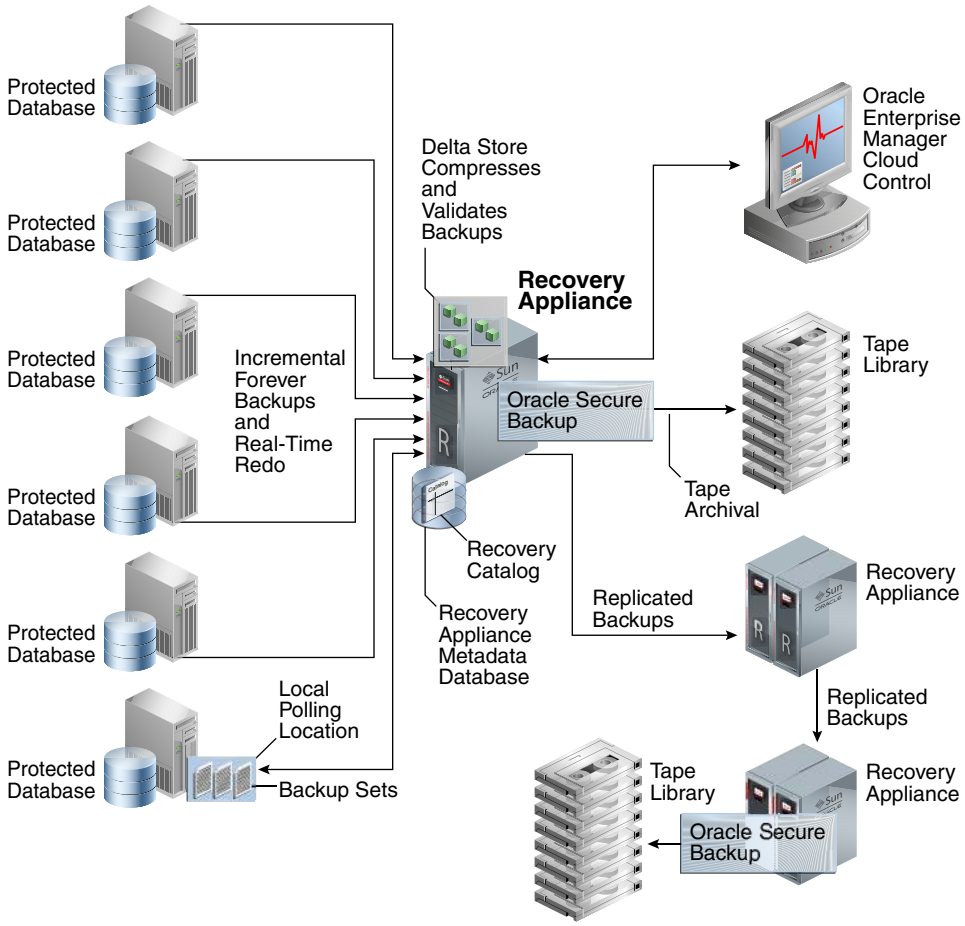
- The Recovery Appliance Environment
- Protected Databases
- Real-Time Redo Transport
- Recovery Appliance Metadata Database
- Recovery Appliance Storage
- Oracle Secure Backup
- Recovery Appliance Replication
- Data Encryption Techniques

2.1 The Recovery Appliance Environment

At minimum, the Recovery Appliance environment consists of one Recovery Appliance and one protected database. More typical is the sample environment shown in Figure 2-1.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Figure 2-1 Sample Recovery Appliance Environment



This section contains the following topics:

- Main Components of the Recovery Appliance Environment
- User Accounts in the Recovery Appliance Environment
- Lifecycle of a Backup: Scenario

2.1.1 Main Components of the Recovery Appliance Environment

Figure 2-1 shows an example of a typical Recovery Appliance environment, which contains the following components:

- Multiple protected databases

  Each protected database sends backups and real-time redo to the Recovery Appliance. Protected databases can run on different releases of Oracle Database.

- Recovery Appliance

  Figure 2-1 shows a central Recovery Appliance, which receives incremental backups and real-time redo from the protected databases. The Recovery Appliance contains the Recovery Appliance metadata database. This database includes the following components:

Copyright © 2014, 2026, Oracle and/or its affiliates.

  - The RMAN recovery catalog, which is subdivided into multiple virtual recovery catalogs.
  - One or more storage locations. Recovery Appliance storage contains the delta store, which includes multiple delta pools.

  Figure 2-1 also shows the central Recovery Appliance replicating backups to a second Recovery Appliance, which in turn forwards these backups to a third Recovery Appliance.

- Oracle Enterprise Manager Cloud Control (Cloud Control)

  Figure 2-1 shows Cloud Control running on a separate server in the environment. Administrators can use Cloud Control to manage all Recovery Appliances, protected databases, and tape devices in the Recovery Appliance environment.

- `DBMS_RA` PL/SQL package

  This is the command-line interface to Recovery Appliance. This package, which is stored in the Recovery Appliance metadata database, provides the underlying functionality for Cloud Control.

- Oracle Secure Backup

  Figure 2-1 shows the Recovery Appliance using Oracle Secure Backup to archive backups to a tape library. The diagram also shows a downstream Recovery Appliance archiving backups to a separate tape library.

> **Note**
>
> Oracle Secure Backup (OSB) 19.1 desupport was announced on May 1, 2026 with premier support end date on September 30, 2027. See My Oracle Support note PNEWS3035 for more information. Please refer to Lifetime Support Policy: Oracle Technology Products for updated support timeframes.
>
> Oracle recommends that database backups to cloud take advantage of Oracle Database Zero Data Loss Cloud Protect, which uses Oracle Zero Data Loss Autonomous Recovery Service in OCI. Cloud Protect offers an efficient incremental forever backup strategy, real-time transaction protection, logically air-gapped immutable backups, and fast, point-in-time recovery.
>
> For customers who require database backups to Amazon S3, Oracle Database Cloud Backup for Amazon S3 may be used. This offering is similar to Oracle Secure Backup Cloud Module in that it is fully integrated with Recovery Manager (RMAN) and enables you to back up your Oracle database to Amazon S3. You can exchange your Oracle Secure Backup license for an Oracle Database Cloud Backup for Amazon S3 license on a 1:1 basis. See Oracle Database Cloud Backup for Amazon S3 for more information.
>
> As with Oracle Secure Backup Cloud Module, Oracle Database Cloud Backup for Amazon S3 can be used free of charge on Oracle Zero Data Loss Recovery Appliance (ZDLRA) for the sole purpose of backing up data to and restoring data from S3-compatible storage devices that are in the same campus as ZDLRA.

Copyright © 2014, 2026, Oracle and/or its affiliates.

See Also

- "Recovery Appliance Metadata Database"
- "Recovery Appliance Storage"
- "Recovery Appliance Replication"
- "DBMS_RA Package Reference"

2.1.2 User Accounts in the Recovery Appliance Environment

The central components of a Recovery Appliance environment are the protected databases, Recovery Appliance, and Cloud Control. Table 2-1 summarizes the most important user accounts in the environment.

Table 2-1 User Accounts in the Recovery Appliance Environment

Component	Account Type	User Name	Description
Cloud Control	Cloud Control super-user	SYSMAN	This application account exists by default. Its purpose is to administer Cloud Control itself, and is not directly related to managing a Recovery Appliance or protected databases.
Cloud Control	Cloud Control administrator	User-Specified	A Cloud Control user account that has been granted the roles and privileges needed to manage a specific protected database or a specific Recovery Appliance. Multiple Cloud Control administrative accounts may exist, depending on your business requirements.
Recovery Appliance	Recovery Appliance metadata database super-user	SYS	SYS can create Recovery Appliance user accounts, but typically is not otherwise used for managing Recovery Appliance.
Recovery Appliance	Recovery Appliance administrator	RASYS	This database account owns the Recovery Appliance schema, which includes the RMAN recovery catalog and the DBMS_RA PL/SQL package (see DBMS_RA Package Reference). The RASYS user name is fixed and cannot be changed. RASYS does not have the privileges required to create database user accounts.

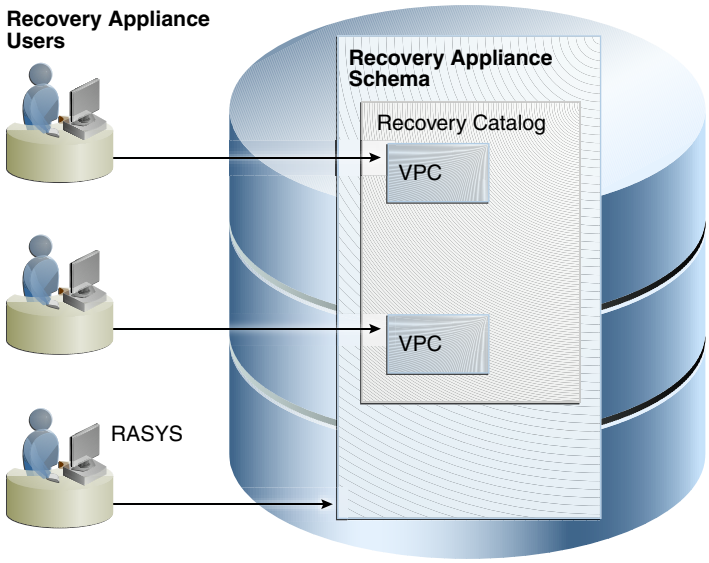
Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 2-1 (Cont.) User Accounts in the Recovery Appliance Environment

Component	Account Type	User Name	Description
Recovery Appliance	Recovery Appliance User Account	User-Specified	This account has authority to send and receive backups for databases registered with the Recovery Appliance, and to manipulate recovery catalog metadata for these databases. This is also the account to use to send redo data from a protected database to the Recovery Appliance. Unlike RASYS, a Recovery Appliance user account has no administrative capabilities in the Recovery Appliance. Typically, a Recovery Appliance metadata database contains multiple Recovery Appliance user accounts. These accounts are created when configuring access for protected databases (see Configuring Recovery Appliance for Protected Database Access). Every Recovery Appliance user account owns a virtual private catalog. The catalog owner can access and modify only those rows in the recovery catalog that pertain to the databases to which it has been granted access. The catalog user name for this is referenced in an RMAN `CONNECT CATALOG` command.
Protected Database	Protected database backup administrator	User account with SYSBACKUP privileges (or SYSDBA for releases in which SYSBACKUP is not supported)	This account has the privileges to back up, restore, and recover a protected database. This is the database user name that is referenced in an RMAN `CONNECT TARGET` command.

Figure 2-2 depicts the relationship between RASYS and two Recovery Appliance user accounts. In this example, each Recovery Appliance user account owns a separate virtual private catalog. Note that RASYS, as owner of the Recovery Appliance schema, is also the owner of the RMAN recovery catalog.

Figure 2-2 RASYS and Recovery Appliance User Accounts



Copyright © 2014, 2026, Oracle and/or its affiliates.

> **See Also**
>
> *Oracle Database Security Guide* to learn how to create database user accounts

2.1.3 Lifecycle of a Backup: Scenario

This section describes the lifecycle of a backup as it flows through the Recovery Appliance environment depicted in Figure 2-1. In this sample scenario, each protected database has already seeded Recovery Appliance with the required initial level 0 incremental backup. The basic data flow is as follows:

1. A protected database, or a standby database protecting this database, sends a level 1 incremental backup to the Recovery Appliance.

   Recovery Appliance distinguishes itself from other backup solutions because only one level 0 backup is ever required for each data file. Level 1 incremental backups are most efficient because data blocks are only backed up when they change.

   Oracle recommends making *cumulative* level 1 incremental backups (see *Oracle Database Backup and Recovery Reference*). Each cumulative level 1 backup uses the most recent *virtual* level 0 backup as its baseline. Typically, this virtual level 0 backup corresponds to the most recent level 1 backup.

   > **Note**
   >
   > If a level 1 cumulative backup cannot be incorporated into the Recovery Appliance (for example, because of a storage corruption), then the next level 1 backup has the same virtual level 0 backup baseline, enabling the Recovery Appliance to seamlessly incorporate the new level 1 incremental backup. Thus, cumulative backups almost never have greater overhead than differential backups.

2. The Recovery Appliance receives the incremental backup.

   The received backup is available for immediate retrieval, but the Recovery Appliance has not yet indexed it, so the corresponding virtual full backup are not available. If a protected database requires this backup for recovery before the Recovery Appliance can index it, then RMAN automatically restores the previous virtual full backup and applies this incremental backup to it.

3. The Recovery Appliance processes the incremental backup.

   The following operations occur asynchronously:

   - The Recovery Appliance performs backup ingest. The Recovery Appliance processes the backup as follows:
     - Scans the backup that was sent by a protected database
     - Breaks it into smaller groups of blocks, assigning the blocks from each data file to a separate delta pool
     - Writes the groups into the appropriate storage location according to the protection policy for the database
     - Deletes the original backup set after the virtual backup set has been created

Copyright © 2014, 2026, Oracle and/or its affiliates.

> **Note**
>
> The Recovery Appliance may not delete the original backup at precisely the same time that the virtual backup is created. Thus, it is possible for both the original and virtual backups to coexist briefly in the recovery catalog as two separate copies.

During backup ingest, the Recovery Appliance also indexes the backup, which involves storing information about the contents and physical location of each data block in the metadata database. Because the Recovery Appliance contains the recovery catalog for the protected database, the newly indexed virtual full backups are now available for use by RMAN, if needed for recovery.

- If Recovery Appliance replication is configured, then the Recovery Appliance forwards the backup to a downstream Recovery Appliance.

  Many different replication configurations are possible. Figure 2-1 shows a one-to-one configuration in which the central Recovery Appliance, acting as the upstream Recovery Appliance (backup sender), forwards its backups to a separate Recovery Appliance, acting as the downstream Recovery Appliance (backup receiver). Figure 2-1 shows cascaded replication, in which the downstream Recovery Appliance forwards its backups to a third Recovery Appliance.

- If automated copy-to-tape policies are enabled, then the Recovery Appliance archives the backup to tape.

  In Figure 2-1, the central Recovery Appliance uses Oracle Secure Backup software to communicate with a tape device. Also, the Recovery Appliance furthest downstream in the replication scheme archives its backups to tape. This technique has the following benefits:

  – To create redundancy, identical backups reside on two separate tape devices. In Figure 2-1, the primary Recovery Appliance archives to tape, as does the Recovery Appliance that is furthest downstream.

  – A downstream Recovery Appliance can back up to tape, thus offloading tape archival processing from the upstream Recovery Appliance.

- The Recovery Appliance periodically verifies that backups and redo are valid.

  The Recovery Appliance automatically validates backups on disk, and during inbound and outbound replication. The Recovery Appliance automatically performs crosschecks of tape backups. Just as with data file backups, the Recovery Appliance validates the integrity of redo log blocks during every operation, including receiving redo from protected databases and storing it in compressed archived log backup sets. No manually run RMAN `VALIDATE` commands are required.

- The Recovery Appliance performs automated delta pool space management.

  This phase involves deleting obsolete and expired backups, both on disk and tape, and optimizing the delta pools.

Copyright © 2014, 2026, Oracle and/or its affiliates.

See Also

- "Recovery Appliance Storage Locations"
- "Replicating Backups with Recovery Appliance "
- "Automated Delta Pool Space Management"
- *Oracle Database Backup and Recovery User's Guide* to learn how to make incremental backups

2.2 Protected Databases

A protected database uses a specific Recovery Appliance as a destination for centralized RMAN backup and recovery. In Figure 2-1, multiple protected databases send backups to a single centralized Recovery Appliance. Each database protected by a Recovery Appliance must use the recovery catalog in the Recovery Appliance metadata database.

To send backups to a Recovery Appliance, a protected database must be configured to allow access to the Recovery Appliance. The configuration involves creating the appropriate Recovery Appliance users and permissions, associating each protected database with a protection policy, and distributing Recovery Appliance connection credentials to each database.

This section contains the following topics:

- Recovery Appliance Backup Modules
- Protection Policies
- Supported Oracle Database Releases

2.2.1 Recovery Appliance Backup Modules

The Zero Data Loss Recovery Appliance Backup Module (Recovery Appliance Backup Module) is an Oracle-supplied SBT library that RMAN uses to transfer backup data over the network to the Recovery Appliance. An SBT library transfers data to and from a backup device type, either a tape device or Recovery Appliance. RMAN performs all backups to the Recovery Appliance, and all restores of complete backup sets, by means of this module.

The Recovery Appliance Backup Module must be installed in the following locations:

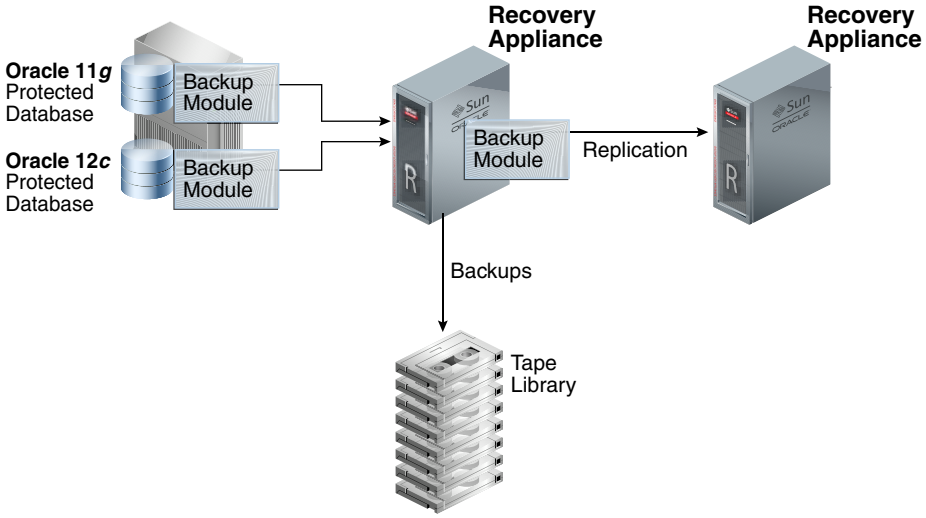
- In the Oracle home of every protected database that sends backups to a Recovery Appliance

  For example, a single host might have an Oracle Database 11*g* Oracle home, and an Oracle Database 12*c* Oracle home. Each Oracle home might support five protected databases, for a total of ten databases running on the host. In this case, only two Recovery Appliance Backup Modules must be installed: one in each Oracle home.
- For Recovery Appliance replication environments, on every upstream Recovery Appliance that sends backups to downstream Recovery Appliances (see Replicating Backups with Recovery Appliance)

Figure 2-3 depicts an Oracle Database 11*g* and Oracle Database 12*c* protected database running on the same host. The Recovery Appliance Backup Module installed in each Oracle home communicates with the Recovery Appliance, replicates backups to a downstream Recovery Appliance.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Figure 2-3 Recovery Appliance Backup Modules



> **See Also**
>
> - *Zero Data Loss Recovery Appliance Protected Database Configuration Guide* to learn how to install the Recovery Appliance Backup Module
> - *Oracle Database Backup and Recovery User's Guide* to learn more about SBT channels and devices

2.2.2 Protection Policies

A protection policy is a named collection of properties that you can assign to multiple protected databases. Using a single policy for multiple databases reduces Recovery Appliance administration time, and enables you to change the properties of multiple protected databases with one operation. To accommodate databases with differing backup and recovery requirements, create as many protection policies as required.

A default installation of Recovery Appliance has the protection policies shown in Table 2-2.

Table 2-2 Default Protection Policies

Service Tier	Recovery Window	Additional Settings
Platinum	45 days on disk, 90 days on tape[1]	Database backups, real-time redo transport, replication, and tape backups. All settings are mandatory.
Gold	35 days on disk, 90 days on tape	Database backups, real-time redo transport, replication, and tape backups (if tape is available).
Silver	10 days on disk, 45 days on tape	Database backups, real-time redo transport, and tape backups (if tape is available).
Bronze	3 days on disk, 30 days on tape	Database backups, and tape backups (if tape is available). There is *no* real-time redo transport.

Copyright © 2014, 2026, Oracle and/or its affiliates.

[1] Backups aged 45 days or less exist on both disk and tape, but backups aged more than 45 days exist only on tape. The Recovery Appliance creates tape backups immediately after disk backups, so the 90 day tape retention period begins at the same time as the 45 day disk retention period.

> **See Also**
>
> *Zero Data Loss Recovery Appliance Protected Database Configuration Guide* to learn how to configure real-time redo transport

2.2.2.1 Protection Policy Attributes

A protection policy is created with the `DBMS_RA.CREATE_PROTECTION_POLICY` procedure or with Cloud Control. The protection policy sets some of the following attributes for all protected databases assigned to it: Some attributes are mutually exclusive. The following is a representative list of attributes to consider in new protection policies.

Table 2-3 Protection Policy Attributes (subset)

Attribute	Description
`storage_location_name`	A Recovery Appliance storage location for storing backups.
`polling_policy_name`	An optional backup polling policy that determines whether Recovery Appliance polls a storage location for backups
`recovery_window_goal`	The disk recovery window goal for the protected database.
`recovery_window_sbt`	The SBT retention period for the protected database.
`guaranteed_copy`	The guaranteed copy setting, which determines whether backups protected by this policy must be copied to tape or cloud before being considered for deletion.
`allow_backup_deletion`	Setting this to `NO` will prevent RMAN users from deleting backups on the Recovery Appliance, necessary for compliance rules. The default value is set to `YES`.
`store_and_forward`	The setting for the Backup and Redo Failover feature. This setting is used only in a protection policy defined on the alternate Recovery Appliance where the protected databases associated with this policy will redirect backups and redo in the event of an outage on the primary Recovery Appliance.
`max_retention_window`	The maximum length of time that the Recovery Appliance retains backups for databases that use this retention policy.
`unprotected_window`	The maximum acceptable difference between the current time and the latest time that the database can be restored.
`autotune_reserved_space`	This setting is used to control whether the Recovery Appliance will automatically define and update the `reserved_space` settings for databases associated with this policy.
`recovery_window_compliance`	This setting specifies a time range for each database backup in which backups will not be deleted. This value must be equal to or smaller than `recovery_window_goal`. Too large a value can result in filling `disk_reserved_space` with compliance protected backups, whereby new backups are then rejected.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 2-3 (Cont.) Protection Policy Attributes (subset)

Attribute	Description
keep_compliance	This setting prevents an administrator from using RMAN CHANGE command to shrink the "keep until time" specified for an archival backup. If KEEP_COMPLIANCE is YES, KEEP FOREVER backups will never be deleted. NO means the "keep until time" for an archival backup may be modified by the RMAN CHANGE command. NO is the default.
max_reserved_space	The maximum disk_reserved_space setting permitted for each database in the protection policy. The format of this value is a character string that must contain a number consisting only of the characters 0-9, followed optionally by one of the following unit specifiers: If max_reserved_space is specified as NULL, the max_reserved_space setting for databases defaults to 2 x disk_reserved_space.
secure_mode	Determines whether backups stored on the Recovery Appliance must be encrypted. YES means that only encrypted backup and redo are accepted by the Recovery Appliance. NO means unencrypted backups are allowed to be stored on the Recovery Appliance. NO is the default.
level0_refresh	If specified, the Recovery Appliance chooses some number of data files from each backup to be level 0 backups. This spreads the creation of new level 0 backup data across the level0_refresh interval. Specify the refresh cycle as any valid INTERVAL DAY TO SECOND expression, such as INTERVAL '20' DAY (20 days). If you set the value to 100 days, then 1% of the database will perform a level 0 backup each day. Effectively, once 100 days is complete, all datafiles will be level 0 with RA_FORMAT=TRUE (or *Block Pool Formatted*). The purpose of this option is to limit the number of Data Encryption Key (DEK) hashes required to restore a database. Each level 1 backup has a new DEK. During a restore, every block is sent to the client along with all the DEKs that are associated with them. During ingest, the DEKs are counted. When 65% of the maximum DEKs for the buffer is reached, a new level 0 refresh is set for that datafile.

You can associate an optional replication server configuration with a protection policy. The replication configuration applies to all protected databases associated with the protection policy.

When a protection policy has SECURE_MODE set to YES, then backups that are not encrypted are rejected before they can be uploaded to the Recovery Appliance, by design. When redo logs are being shipped directly to the Recovery Appliance, they also must be encrypted. However, the check for redo encryption happens *after* the redo log completes, so future attempts to open a new log on the Recovery Appliance are rejected. A few logs might get started before the archived log destination status shows redo being rejected. This condition clears when an encrypted redo log backup is sent to the Recovery Appliance. After which, future redo log switch are accepted on the Recovery Appliance.

Copyright © 2014, 2026, Oracle and/or its affiliates.

> **Note**
>
> Before release 21.1, any backup copy anywhere (tape or cloud) counted as a copy for a backup and would allow for deletion on the Recovery Appliance. If you had both cloud and tape, you might have incomplete backups on either cloud and tape, but the Recovery Appliance would incorrectly consider the set copied. Further with replication, the backups could be deleted on the downstream Recovery Appliance, leave backups never copied, and thus never released by the upstream Recovery Appliance.
>
> After release 21.1, the `guaranteed_copy` attribute was added to the library. When `guaranteed_copy` is set on the library, the Recovery Appliance will not directly delete the copy in the library. [The tape/cloud manager shouldn't delete the copy either.] Each library with the `guaranteed_copy` attribute must have a copy of a given backup before it is eligible for deletion from the Recovery Appliance.
>
> The APIs `create_protection_policy` and `update_protection_policy` check whether a `guaranteed_copy` library/template/`attribute_set` was available to the `protection_policy` before the `protection_policy` could have `guaranteed_copy` set. Other improvements protect the changing of libraries, templates, or `attribute_set` against the last removal of a library/template/`attribute_set` path from a `protection_policy` with the `guaranteed_copy` attribute set.

2.2.2.2 Recovery Windows

When creating a protection policy, you can define the following two recovery window attributes, expressed as intervals (typically days):

- Disk recovery window goal

  For each database assigned to the policy, Recovery Appliance attempts to support a point-in-time recovery to any time within this interval, counting backward from the current time. For example, if the recovery window goal is 15 days, and if it is noon on April 25, then the goal is the ability to perform point-in-time recovery to any time on or after noon on April 10. At noon on April 26, the goal is the ability to perform point-in-time recovery to any time on or after noon on April 11, and so on.

  For disk, this interval is a goal, and not a guarantee. The Recovery Appliance might purge backups when disk space is low, in which case the goal is not always met. You can ensure that a minimum number of backups are guaranteed to be available by adjusting the reserved disk space property of each protected database.

- SBT retention period

  For each assigned database, backups are retained long enough on tape to support a point-in-time recovery to any time within this interval, counting backward from the current time. For SBT, this interval is a guarantee.

Copyright © 2014, 2026, Oracle and/or its affiliates.

> **See Also**
>
> - "Recovery Window Goal"
> - ""Backup Retention on Tape""
> - *Zero Data Loss Recovery Appliance Protected Database Configuration Guide*
> - *Oracle Database Backup and Recovery User's Guide* for a thorough discussion of recovery windows

2.2.2.3 Backup Polling Policies

A backup polling policy specifies:

- A file system directory on shared storage where Recovery Appliance polls for backups to process (see "Backup Polling Locations")
- The frequency with which Recovery Appliance polls
- Whether backup data is to be deleted after being successfully processed

Assign backup polling policies to protected databases through protection policies. Each protection policy can optionally reference a polling policy.

> **See Also**
>
> "Creating a Backup Polling Policy (Command-Line Only)"

2.2.3 Supported Oracle Database Releases

See My Oracle Support Note Doc ID KB439308 for information about the Oracle database releases supported by Recovery Appliance, including the features available with each release.

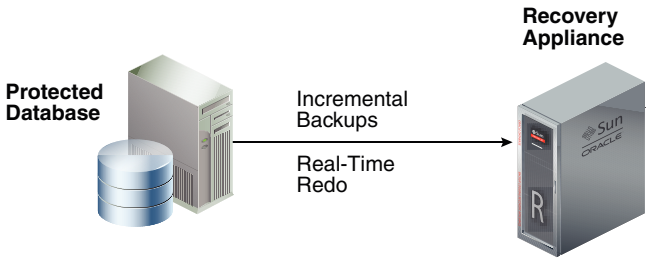
2.3 Real-Time Redo Transport

Redo data contains records of all changes made to a database and is therefore critical to minimizing data loss if data failure occurs. By using the real-time redo transport feature of Recovery Appliance, you substantially reduce the window of potential data loss that exists between successive archived redo log backups. Typical RPO is zero to subsecond when you enable real-time redo transport.

Figure 2-4 shows a protected database sending incremental backups and redo logs to the Recovery Appliance.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Figure 2-4 Redo Log Transmission



With real-time redo transport enabled, a protected database generates redo changes in memory, and then immediately transfers them to the Recovery Appliance, which validates them and writes them to a staging area.

When the protected database performs an online redo log switch, the Recovery Appliance converts and assembles the redo changes into compressed archived redo log file backups. The Recovery Appliance catalog automatically tracks these archived redo log backups in its recovery catalog. RMAN can restore and apply these archived redo log backups as usual. The advantages are:

- If the redo stream terminates unexpectedly, then the Recovery Appliance can close the incoming redo stream and create a partial archived redo log file backup, thereby protecting transactions up to the last change that the appliance received. When the Recovery Appliance detects that the redo stream has restarted, it automatically retrieves all missing archived redo log files from the protected database. In this way, the Recovery Appliance can preserve the recovery window goal.
- Because the Recovery Appliance automatically converts real-time redo into archived redo log files, it is not necessary to back up archived redo log files from the database host to the Recovery Appliance.

The Recovery Appliance does not *apply* the redo that it receives to the backups sent by the protected databases. Thus, to continue providing updated virtual level 0 backups, the Recovery Appliance must incorporate new incremental backups into the delta store. The appliance provides a virtual level 0 backup corresponding to each level 1 incremental backup sent by the protected database. In a recovery scenario, you restore the appropriate level 0 backup, and then use redo log files to roll it forward.

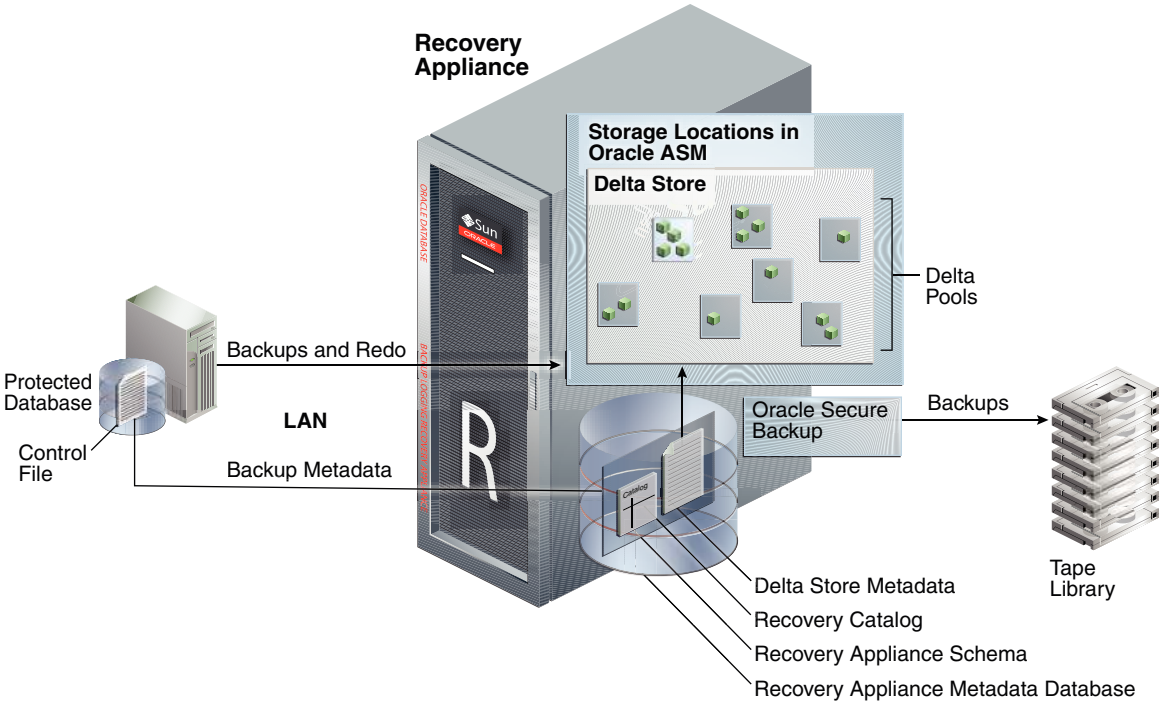
> **See Also**
>
> *Zero Data Loss Recovery Appliance Protected Database Configuration Guide* to learn more about real-time redo transport and how to enable it

2.4 Recovery Appliance Metadata Database

The key component of the Recovery Appliance is the Recovery Appliance metadata database. This database manages metadata for all backups, and also contains the RMAN recovery catalog. The Recovery Appliance metadata database is preconfigured, pretuned, and managed by the Recovery Appliance. Figure 2-5 depicts a Recovery Appliance metadata database interacting with a protected database.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Figure 2-5 Recovery Appliance Metadata Database



This section contains the following topics:

- Delta Store
- Delta Pools
- Automated Delta Pool Space Management
- Recovery Appliance Schema
- Recovery Appliance Catalog

2.4.1 Delta Store

The delta store is the totality of all protected database backup data in Recovery Appliance storage location. All data file and archived redo log backups reside in the delta store. The delta store contains delta pools for all data files in all protected databases.

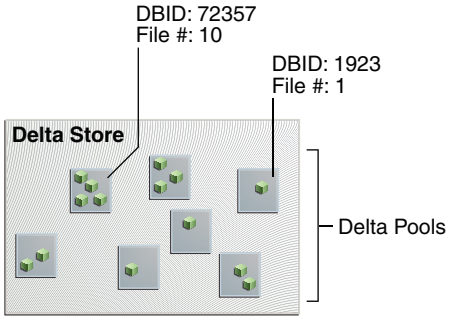
2.4.2 Delta Pools

The delta store is the collection of delta pools. As the Recovery Appliance receives backups from protected databases, it indexes them and stores them in delta pools. A delta pool is the set of data file blocks from which Recovery Appliance constructs virtual full backup. Recovery Appliance automatically manages delta pools so that it can create a virtual full backup that corresponds to any incremental backup ever received.

Each separate data file whose backups are sent to Recovery Appliance has its own separate delta pool. For example, data file `10` from `prod1` has its own delta pool, data file `1` from database `prod2` has its own delta pool, and so on. As shown in Figure 2-6, the delta store contains all the delta pools for the databases protected by Recovery Appliance.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Figure 2-6 Delta Pools in Delta Store



2.4.3 Automated Delta Pool Space Management

The set of operations by which Recovery Appliance manages backups is called automated delta pool space management. Specifically, space management involves the following automated tasks:

- Deleting backups (both in a Recovery Appliance storage location and on tape) that are obsolete or expired based on the disk recovery window goal and SBT retention policy

  Recovery Appliance periodically determines that some backups no longer need to be stored on disk, so their disk space can be reclaimed. When the Recovery Appliance determines that some backups residing in the delta pools are obsolete, the individual blocks that compose those backups are typically located in physical files alongside non-obsolete blocks. Recovery Appliance rewrites these physical files so that the delta pools can reclaim the space occupied by the obsolete blocks.

- Reorganizing the delta pools periodically to improve performance of restore operations

  The automatic tracking and reorganizing of the delta pools is called delta pool optimization. As old blocks are deleted and new incremental backups arrive for updated data files, the blocks in a backup can become less contiguous. This state can degrade the performance of restore operations. Recovery Appliance runs a background task that automatically reorganizes virtual full backup blocks to maintain contiguity, thus optimizing read access for restore operations.

> **See Also**
>
> "How Recovery Appliance Manages Storage Space"

2.4.4 Recovery Appliance Schema

The Recovery Appliance schema contains metadata used internally by the Recovery Appliance to manage backups on behalf of its protected databases. RASYS is the Recovery Appliance administrative user who owns the Recovery Appliance schema. The Recovery Appliance schema contains the RMAN recovery catalog.

Copyright © 2014, 2026, Oracle and/or its affiliates.

See Also

"User Accounts in the Recovery Appliance Environment"

2.4.5 Recovery Appliance Catalog

Updates to the recovery catalog reflect the results of Recovery Appliance indexing and space management collection. These updates do not occur in the control files of the protected databases. For this reason, protected databases that store backups in the Recovery Appliance must use the Recovery Appliance catalog.

Note

Protected databases may use the recovery catalog in the Recovery Appliance without also using the Recovery Appliance as their backup repository.

RMAN connects to the Recovery Appliance catalog using the same Recovery Appliance account employed for backup and recovery operations. Each Recovery Appliance user account is also a virtual private catalog account. The `DBMS_RA.GRANT_DB_ACCESS` procedure grants Recovery Appliance privileges to a database user account for a specified protected database.

See Also

- "Enrolling Protected Databases"
- "GRANT_DB_ACCESS"
- *Oracle Database Backup and Recovery User's Guide* to learn how to manage a recovery catalog

2.5 Recovery Appliance Storage

Recovery Appliance uses the following types of storage:

- Recovery Appliance storage location

  This Oracle ASM location is the main storage for backups on Recovery Appliance disks, serving as the destination for protected database backups.

- Backup polling location

  An optional file system directory on shared storage, outside the Recovery Appliance, that is a destination for backup pieces and archived redo log files from a protected database. Recovery Appliance polls the directory at specified intervals, retrieves any found backups, and then processes and stores them.

Copyright © 2014, 2026, Oracle and/or its affiliates.

See Also

- "Recovery Appliance Storage Locations"
- "Backup Polling Locations"
- *Zero Data Loss Recovery Appliance Protected Database Configuration Guide* to learn how to configure real-time redo transport

2.5.1 Recovery Appliance Storage Locations

A Recovery Appliance storage location can be shared among multiple protected databases. The Recovery Appliance administrator decides which clients will use each storage location.

2.5.1.1 Benefits of Recovery Appliance Storage

The benefits of Recovery Appliance storage locations are:

- More efficient disk usage

  Recovery Appliance uses common storage to absorb spikes from all protected databases, reducing the total amount of over-allocated storage. In traditional RMAN backup and recovery, a fast recovery area stores recovery-related files. Individual fast recovery areas require that each database maintain the amount of storage required to accommodate its largest expected activity spike, which often results in wasted storage.

  Note

  The default storage location in the Recovery Appliance also contains a fast recovery area for catalog backups.

  Oracle recommends that protected databases continue to maintain fast recovery areas for storage of local online and archived redo log files, control file autobackups, and flashback logs. In a Recovery Appliance environment, the fast recovery areas have smaller space requirements because RMAN backups are stored in the Recovery Appliance.
- Database-optimized backup deduplication and compression
- Shared disk backup pool distributed based on database protection policy, which defines the disk recovery window goal for each database protected by the policy

2.5.1.2 Oracle ASM and Recovery Appliance Storage

Recovery Appliance storage locations occupy space in Oracle ASM disk groups. By default, the delta pool is stored in normal redundancy Oracle ASM disk groups, which means that the Recovery Appliance maintains two copies of all on-disk backups. Database backups can survive the loss of any one disk or storage server. The Recovery Appliance metadata database, which tracks the files and blocks, is stored in a high redundancy Oracle ASM disk group.

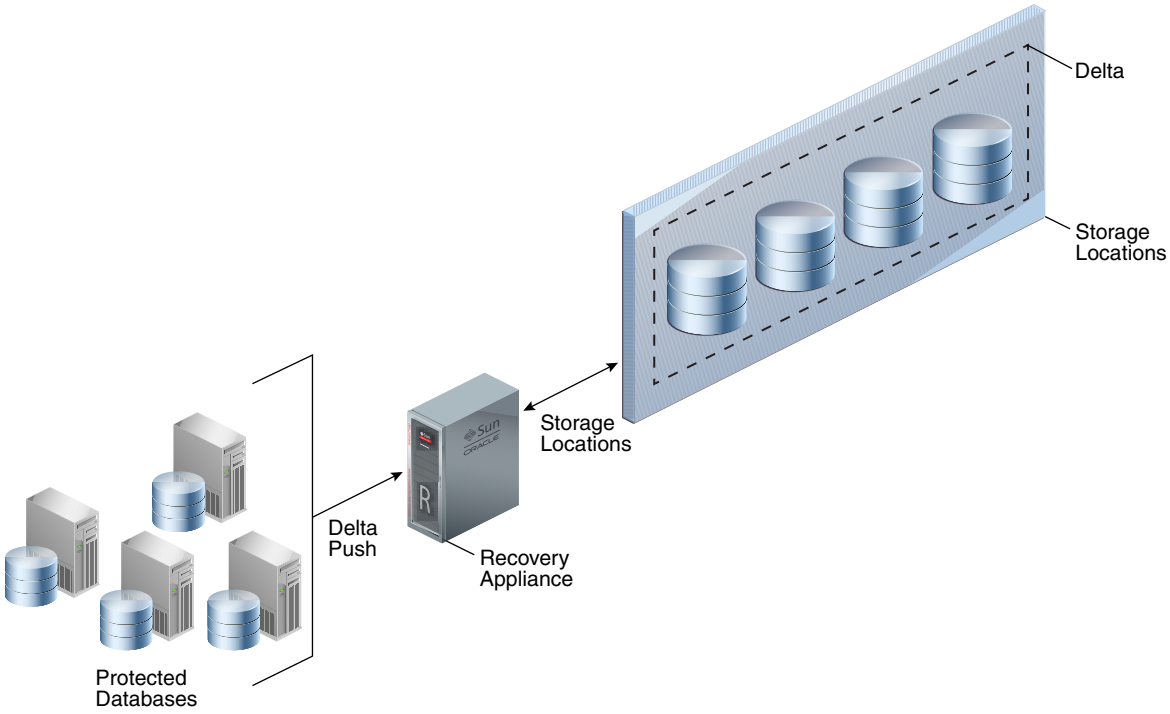
Copyright © 2014, 2026, Oracle and/or its affiliates.

> **See Also**
>
> - "How Recovery Appliance Manages Storage Space"
> - *Oracle Automatic Storage Management Administrator's Guide*

2.5.1.3 DELTA Storage Location

By default, Recovery Appliance is configured with all available disk storage assigned to a single storage location called `DELTA`. As shown in Figure 2-7, all protected databases share this storage location.

Figure 2-7 DELTA Storage Location



2.5.2 Backup Polling Locations

A backup polling policy defines a file system directory where a protected database places backups without interacting directly with Recovery Appliance. The backup polling directory is an NFS mount point, and is not in a Recovery Appliance storage server.

The polling policy defines the file system path to the storage and how often it will be searched for new backups. Polling policies are optional and do not need to be created if backups are not sent to Recovery Appliance using the polling method.

Copyright © 2014, 2026, Oracle and/or its affiliates.

> **See Also**
>
> "Backup Polling Policies"

2.5.2.1 Stages of Backup Polling

Backup polling occurs in the following stages:

1. The protected database writes backups without the involvement of Recovery Appliance, which does not need to be running while backups are created.
2. Recovery Appliance polls for newly arrived backups.
3. When Recovery Appliance discovers a file through polling, the Recovery Appliance examines its contents and tries to associate it with a protected database, and then does either of the following:
   - If the file is associated with a registered protected database, then Recovery Appliance processes the backup.
   - If the file is not associated with any registered protected database, then Recovery Appliance logs a warning message and does not process the file.

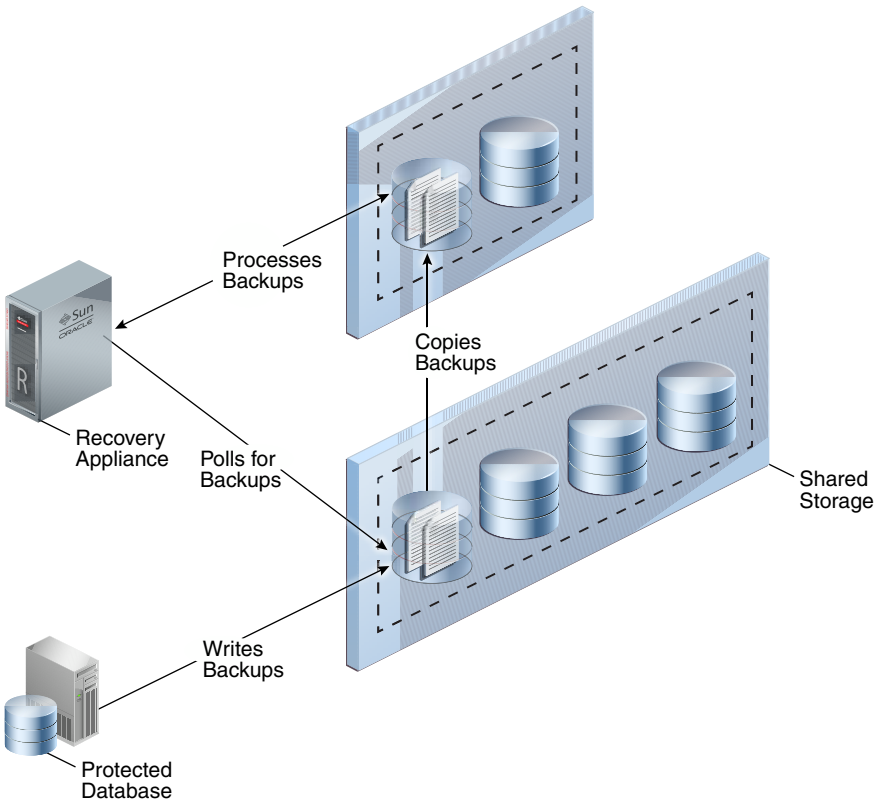
2.5.2.2 How Recovery Appliance Processes Backups in Backup Polling Directories

To set up polling so that backups are copied to Recovery Appliance storage, you must configure a backup polling directory that exists *outside* Recovery Appliance storage locations, but which the Recovery Appliance can access. The protected database writes its backups to the polling directory, which you specify in the polling policy.

The Recovery Appliance checks the polling directory for newly created backups. When backups exist, the Recovery Appliance copies the backups from the polling directory to its internal Recovery Appliance storage location, and then processes them. After enough time has passed for Recovery Appliance to copy the backups, the protected database deletes the backups from the polling directory. Figure 2-8 depicts this configuration.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Figure 2-8 Backup Polling



2.5.3 How Recovery Appliance Manages Storage Space

An important duty of the Recovery Appliance administrator is planning for the proper amount of disk space for a specified retention window and database size. As conditions change, the Recovery Appliance provides space management monitoring and alerting at the storage location and database level. When estimated storage needs are approaching the amount of available storage, alerts and warnings give the administrator time to accommodate the storage demands.

The following attributes, whose settings are accessible through the RA_DATABASE view, determine how Recovery Appliance manages storage space and backup retention:

- Recovery Window Goal
- Reserved Space
- Guaranteed Copy
- Maximum Retention Window
- Recover Window Compliance

> **See Also**
>
> "Archival and Encrypted Backups" for special algorithms that apply to RMAN backups that are not part of the incremental-forever strategy

Copyright © 2014, 2026, Oracle and/or its affiliates.

2.5.3.1 Recovery Window Goal

The `recovery_window_goal` parameter of `DBMS_RA.CREATE_PROTECTION_POLICY` specifies the interval (typically in days) within which point-in-time recovery must be possible, counting backward from the current time. Consider a `recovery_window_goal` setting of 1 day. At midnight on August 7, the goal is recoverability to any time between the current time and midnight on August 6. At midnight on August 8, the goal is recoverability to any time between the current time and midnight on August 7, and so on.

Recovery Appliance attempts to retain sufficient backups to meet the recovery window goal defined for each database. For example, a Recovery Appliance protects databases `STORE01`, `STORE02`, and `STORE03`. The recovery window goal for `STORE01` is 1 day. If at midnight on August 7, `STORE01` needs 624.2 GB for backups to meet its recovery window goal, then the Recovery Appliance attempts to ensure that at least this much space is allocated for `STORE01` backups.

If sufficient space exists in storage, then backups created before a recovery window goal may be available—although they are not guaranteed. If purging previous backups is not necessary, then the Recovery Appliance keeps them, effectively extending the time to which point-in-time recovery is available. For example, on August 7 the space available to `STORE01` might be 700 GB or more, even though only 624.2 GB is required. A similar situation may exist for `STORE02` and `STORE03`.

If sufficient space does *not* exist in storage, then by default (`guaranteed_copy=NO`) the Recovery Appliance may purge backups. When reclaiming space, the Recovery Appliance attempts to respect the recovery window requirement first.

See Also

- "Creating a Protection Policy"
- "CREATE_PROTECTION_POLICY"

2.5.3.2 Reserved Space

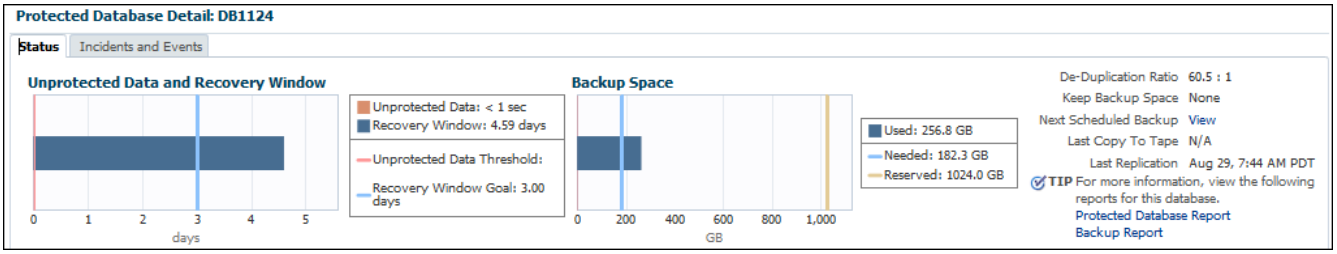
Next in importance to the recovery window goal is the `reserved_space` parameter of `DBMS_RA.ADD_DB` and `DBMS_RA.UPDATE_DB`. The reserved space defines the amount of disk space guaranteed to each protected database to meet its recovery window goal.

Note

This is the only storage parameter that is specified in `ADD_DB` rather than in the protection policy.

Because backups need space, you must estimate how much reserve space you believe is needed to store backups. For example, you might allocate 1024 GB of reserved space to the `DB1124` database, which means the Recovery Appliance guarantees 1024 GB to `DB1124` *if* the database needs this amount to meet its recovery window goal. The following graphic shows a section of a Protected Database Details report:

Copyright © 2014, 2026, Oracle and/or its affiliates.



In the preceding example, the disk recovery window goal for `DB1124` is 3 days, and the actual recovery window (the time to which the Recovery Appliance can currently recover) is 4.59 days. Meeting the recovery window goal requires 182.3 GB of backup data. This amount is less than 20% of the specified reserved space setting of 1024 GB. By default, at any given time, a database may actually have more or less than its specified reserved space available.

> **Note**
>
> Reserved space is measured in space (GB), whereas the recovery window goal is measured in time.

The Recovery Appliance uses recovery window goals and reserved space settings to allocate storage *dynamically* to meet business requirements. If the Recovery Appliance has purged as much backup data as possible while still meeting the recovery window goal for each database, and if more space is needed, then the Recovery Appliance evaluates the reserved space setting of each database. Recovery Appliance purges backups for the database whose backups exceed the reserved space by the highest percentage, and logs a message in the RA_INCIDENT_LOG view. Query the RA_PURGING_QUEUE view to determine which database will next have a backup purged.

The ESTIMATE_SPACE procedure can assist with determining reserved space. When calculating reserved space for a *compliance* protection policy, the `target_window` should be the `RECOVERY_WINDOW_COMPLIANCE` ***plus an extra day*** for edge conditions.

> **See Also**
>
> - "Enrolling Protected Databases"
> - "CREATE_PROTECTION_POLICY"
> - "ADD_DB"

2.5.3.3 Guaranteed Copy

A key question in storage management is whether it is more important to ensure that older backups are copied to tape or cloud than it is to accept new backups or redo. The following settings are possible for the `guaranteed_copy` parameter of `DBMS_RA.CREATE_PROTECTION_POLICY`:

- `NO` (default)

Copyright © 2014, 2026, Oracle and/or its affiliates.

Recovery Appliance can purge backups before they have been copied to tape or cloud when it is necessary to make space for newer backups. In this case, the protected database may have more or less than the reserved space.

- YES

  When guaranteed copy is enabled, Recovery Appliance never purges a backup before it has been copied to tape or cloud. The Recovery Appliance can only hold up to `disk_reserve_space` bytes of backup data that is not yet copied to all libraries with the `guaranteed_copy=YES`.

  After the Recovery Appliance has consumed reserved space with backups that have not been copied to tape or cloud, the Recovery Appliance cannot accept new backups or redo. This setting only changes the behavior of storage management when the tape system or replicated Recovery Appliance is unavailable for an extended period.

  The Recovery Appliance uses a different algorithm for virtual backups that are part of an incremental-forever backup strategy. Non-virtual backups occupy a specific amount of space and either have or do not have a tape copy. For virtual backups, the tape schedule may write either a level 1 or level 0 version of any virtual backup in the Recovery Appliance. Additionally, space computations for virtual backups are complex because the space includes blocks needed to support the backup, which may differ from the space needed to write the backup to tape. For these reasons, after the Recovery Appliance writes a virtual data file backup to tape, the Recovery Appliance considers all versions of this backup and any older virtual backups of this data file as copied to tape (or replicated).

See Also

- "Delta Push"
- "Creating a Protection Policy"
- "CREATE_PROTECTION_POLICY"

2.5.3.4 Maximum Retention Window

The `max_retention_window` parameter of `DBMS_RA.CREATE_PROTECTION_POLICY` specifies the maximum time that the Recovery Appliance retains backups for databases using this policy. Specifying null means that no backup purging occurs unless caused by space pressures within a storage location, or user actions.

The Recovery Appliance only keeps backups longer than the retention window when necessary to preserve the recovery window goal for a database. The effect of this setting is that the Recovery Appliance deletes backups sooner than it might otherwise have chosen to delete them.

See Also

- "Creating a Protection Policy"
- "CREATE_PROTECTION_POLICY"

Copyright © 2014, 2026, Oracle and/or its affiliates.

2.5.3.5 Recover Window Compliance

The `RECOVERY_WINDOW_COMPLIANCE` parameter of `DBMS_RA.CREATE_PROTECTION_POLICY` specifies for each database using the policy a range of time that backups will not be deleted. These backups must not use more than `disk_reserved_space` bytes of storage, and if they do, new backups will be rejected until those backups age out of the range.

`RECOVERY_WINDOW_COMPLIANCE` is different and more restrictive than `RECOVERY_WINDOW_GOAL`, because the *goal* doesn't have to be met but the *compliance* does. The goal might be for the Recovery Appliance to recover a given database to any point in the last 30 days, if reserve storage is sufficient and not needed and overwritten by newer backups. Recovery window compliance might require the Recovery Appliance to recover a given database to any point in the past 7 days regardless of reserve storage constraints.

Because backups need space, you must estimate how much reserve space you believe is needed to store backups. The ESTIMATE_SPACE procedure can assist with determining reserved space. When calculating reserved space, the `target_window` should be the `RECOVERY_WINDOW_COMPLIANCE` ***plus an extra day*** for edge conditions.

> **Note**
>
> If the `RECOVERY_WINDOW_COMPLIANCE` is too large, it can prevent the addition of new backups to the Recovery Appliance, because reserve storage isn't available. When `RECOVERY_WINDOW_COMPLIANCE` consumption is near the reserved storage limit and an incoming backup piece would have the space used exceed that limit, RMAN fails immediately.

Changes can be made to the protection policy to keep backups longer or shorter for new backups. However, once `RECOVERY_WINDOW_COMPLIANCE` is set for a given backup, it is strictly enforced and the backup is not deleted until the `RECOVERY_WINDOW_COMPLIANCE` period expires.

> **See Also**
>
> - "Creating a Protection Policy"
> - "CREATE_PROTECTION_POLICY"

2.5.3.6 Archival and Encrypted Backups

The following types of backups cannot be part of an incremental-forever strategy, or be used to construct virtual full backups:

- RMAN archival backups created using the `BACKUP ... KEEP` command
- RMAN encrypted backups created using `CONFIGURE` or `SET ENCRYPTION`. `SET ENCRYPTION ON` backups are supported for incremental forever when RMAN channels are allocated with `PARMS=(RA_FORMAT=TRUE)`.

The Recovery Appliance manages the preceding backups differently from backups in an incremental-forever strategy. Recovery Appliance retains archival backups regardless of the specified recovery window goal. However, encrypted backups do adhere to recovery window settings.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Archival backups are eligible for deletion by the Recovery Appliance only after the KEEP time expires. If you intend to store archival backups for an extended time, then note the following guidelines:

- Adjust the reserved space to account for them. Archival backups reduce the space available for achieving your recovery window goal and must be accounted for.
- Because the Recovery Appliance does not automatically copy archival backups to tape, you must manually copy them using the COPY_BACKUP procedure. This procedure also enables you to copy archival backups to disk locations that are outside Recovery Appliance storage locations. The MOVE_BACKUP procedure copies an archival backup to disk or tape and then deletes it from the storage area.

See Also

- "Copying Backups to Tape with Recovery Appliance "
- "Data Encryption Techniques"
- *Oracle Database Backup and Recovery User's Guide* to learn more about archival backups

2.5.3.7 Space-Efficient Encrypted Backups

L0 and L1 incremental backups can be compressed and encrypted in a space-efficient manner, including TDE database backups.

Prior to the availability of this feature with RA 23.1, TDE databases were supported for incremental forever backup to the Recovery Appliance. However, due to the encrypted data format, backup compression performed by the Recovery Appliance did not yield additional savings.

WIth the space-efficient encrypted backup feature in RA 23.1, RMAN working with the Recovery Appliance decrypts the data file blocks using the TDE master key from the same wallet or key store for the protected database. This is followed by compressing the blocks, and then re-encrypting with a new data encryption key. The resulting backup payload is then formatted for virtual full compatibility. Finally, the data encryption key is wrapped with the TDE master key and stored in the backup, before being sent to the Recovery Appliance.

During restore of space-efficient encrypted backups, RMAN works with the Recovery Appliance backup module to decrypt, decompress, and re-encrypt data blocks (if TDE) before writing data files into the database storage destination.

The Recovery Appliance supports space-efficient encrypted backup lifecycle (validation, purging, etc.) without the need to access database keys or decrypt block data. This keeps the Recovery Appliance's role and responsibilities completely separate from the protected database.

Note

Do not purge keys from your wallet or keystore, because old blocks needed in a restore might require old TDE master keys. This is particularly important to consider for backups that are put on external storage.

Space-efficient encrypted backups are only supported on Linux database platforms.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Prior to Oracle Database 23ai, the default encryption algorithm is `AES128`. A different encryption algorithms can be set, which then gets used.

With Oracle Database 23ai and later, the default encryption algorithm is `AES256-XTS` and is configurable.

Refer also to Configuring Space-Efficient, Encrypted Backups

2.6 Ransomware and Cybersecurity

The Oracle Zero Data Loss Recovery Appliance is engineered for database ransomware protection. It has four key technology pillars:

- **Database Protection** includes real-time transaction protection and end-to-end ransomware protection and immutability.
- **Recovery Assurance** includes continuous backup validation, database protection monitoring, as well as high-speed, fast database restore capabilities through a dedicated network.
- **Resilient Architecture** built on a compute and storage servers foundation, which stems from Oracle Exadata engineered systems design methodology. The user model has a separation of duties; the roles for databases, the Recovery Appliance, and for any related appliances are segregated from each other. No one user can access other systems which they are not privileged to do so.

  Immutable Backups prevents the backups themselves on a compromised system to be purged or deleted by internal processes or external users.

The Recovery Appliance has Resiliency and Recoverability from Cyber-Attacks

The Oracle Zero Data Loss Recovery Appliance is designed to be fault-isolated from the production database. If a cyber-attack hits the production database, the Recovery Appliance is not compromised. This solution stems from the following key architectural features:

- **End-to-End Data Validation**

  The Recovery Appliance validates all incoming, on-disk, and replicated backups for Oracle block correctness and recoverability. Any backup data maligned by malware or ransomware attack is detected, recorded, and alerted to the administrator. Action can then be taken in conjunction with the DBAs to disconnect the database from the network and investigate further.

  Replicated backups cannot be deleted or modified by the primary appliance or its administrators. They are independently validated and managed by the replication Recovery Appliance. They are shielded from any effects of attacks done on the primary Recovery Appliance.

- **Air-Gapped Vault Backups**

  With Recovery Appliance database-aware incremental-forever replication, the vault appliance is configured behind a firewall which has a window open only during certain times of the day. Recovery Appliance replication proceeds during those times to synchronize the vault appliance. When the firewall is closed, replication pauses. Upon the next open sync window, replication resumes. With incremental forever-based replication, only the minimum amount of data is required to maintain full recoverability from the vault. Unlike with general purpose storage appliances, no full backups are transmitted, which thus limits the sync window and possibility of malicious access to the vault.

- **Separation of Duty**

Copyright © 2014, 2026, Oracle and/or its affiliates.

Access to the system is controlled via strict separation of duty between DBA and Recovery Appliance administrator roles. DBAs are only given Virtual Private Catalog (VPC) user roles to backup and recover their privileged databases. They cannot access, modify, or delete backups on the Recovery Appliance.

Recovery Appliance administrators only have access to manage and monitor the system, but cannot backup, recover, or modify protected databases. The Recovery Appliance does not expose or allow creation of local users, databases, or other services.

- **Limited Network Access**

  With regards to network protocols, VPC users can only connect to the appliance through SQL*Net. HTTPS is used for RMAN backup and restore traffic through the Recovery Appliance Backup Module. No other protocols are employed.

  The Recovery Appliance enforces network segregation with the support of VLAN tagged networks, allowing backup and restore traffic to be fully isolated and non-routable between protected databases' specific network zones. In this way, any possibly affected backups would not be exposed to the rest of the enterprise.

- **Superior Resiliency**

  As an Oracle Engineered System built on Exadata hardware and storage, the Recovery Appliance inherits a resilient architecture for reducing surface of attack on compute and storage servers. This includes:

  – hardened password policies
  – OS and DB user auditing
  – firewall support
  – Oracle ILOM (Integrated Lights Out Management)

  Recovery With No Data Loss

  In the event that a database server is attacked and its backups must be recovered to a different server, the Recovery Appliance's real-time redo transport allows recovery to the very last transaction prior to the attack occurrence.

2.7 Oracle Secure Backup

Oracle Secure Backup is the tape management component of Recovery Appliance. The Recovery Appliance offloads tape backup operations from protected databases to the Recovery Appliance. Thus, protected database hosts do not need the RMAN-integrated media management software module. Instead, a single copy of the Oracle Secure Backup module is installed on Recovery Appliance. The Recovery Appliance automatically manages the copy of backups to tape for all protected databases.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Note

Oracle Secure Backup (OSB) 19.1 desupport was announced on May 1, 2026 with premier support end date on September 30, 2027. See My Oracle Support note PNEWS3035 for more information. Please refer to Lifetime Support Policy: Oracle Technology Products for updated support timeframes.

Oracle recommends that database backups to cloud take advantage of Oracle Database Zero Data Loss Cloud Protect, which uses Oracle Zero Data Loss Autonomous Recovery Service in OCI. Cloud Protect offers an efficient incremental forever backup strategy, real-time transaction protection, logically air-gapped immutable backups, and fast, point-in-time recovery.

For customers who require database backups to Amazon S3, Oracle Database Cloud Backup for Amazon S3 may be used. This offering is similar to Oracle Secure Backup Cloud Module in that it is fully integrated with Recovery Manager (RMAN) and enables you to back up your Oracle database to Amazon S3. You can exchange your Oracle Secure Backup license for an Oracle Database Cloud Backup for Amazon S3 license on a 1:1 basis. See Oracle Database Cloud Backup for Amazon S3 for more information.

As with Oracle Secure Backup Cloud Module, Oracle Database Cloud Backup for Amazon S3 can be used free of charge on Oracle Zero Data Loss Recovery Appliance (ZDLRA) for the sole purpose of backing up data to and restoring data from S3-compatible storage devices that are in the same campus as ZDLRA.

See Also

Oracle® Database Backup and Recovery Reference.

2.7.1 Tape Archival

Protected databases send backups to the Recovery Appliance, which stores them on disk in the specified storage locations. To reclaim disk space and to create transportable tape backups, business requirements may necessitate archival to tape.

Tape backups are a repetitive task that the Recovery Appliance automates and performs as a background task. The Recovery Appliance administrator configures rules that specify the frequency with which the Recovery Appliance creates tape backups. Because the protection policy is a natural grouping of databases, databases sharing the same protection policy can share the same tape archival requirements.

Recovery Appliance creates tape backups in the compatibility version that matches the greatest compatibility version of any full or incremental backup that contributes blocks to the backup. The database identity information that the Recovery Appliance sends to the media management layer is identical to the information that would be sent if the database were sending the backup. This consistency guarantees that the database for which the backups were created can use them.

Copyright © 2014, 2026, Oracle and/or its affiliates.

2.7.2 Tape Retrieval

Backups are stored on tape as complete backup sets, not virtual backups, so tape backups are usable by RMAN without mediation by the Recovery Appliance. You can restore backups from tape in the following ways:

- Retrieval by Recovery Appliance

  This is the simplest way to restore an SBT backup created by Recovery Appliance. RMAN requests the restore from Recovery Appliance, without needing to be aware that this particular backup has been moved to tape. Recovery Appliance recognizes that the requested backup set is located on tape, open an SBT session to restore the backup from the media manager, and transfer the data over the network to the protected database host.

- Retrieval by protected database

  Because the SBT backups exist in a client-compatible format, RMAN can restore the backups from tape directly to any host, without involving Recovery Appliance. In this case, RMAN must first catalog the backup pieces before it can restore them from tape, and the Oracle Secure Backup library must be installed on the protected database host.

> **See Also**
>
> *Oracle Database Backup and Recovery User's Guide* to learn how to restore backups from tape

2.8 Recovery Appliance Replication

In Recovery Appliance replication, one Recovery Appliance (the upstream Recovery Appliance) forwards backups to another Recovery Appliance (the downstream Recovery Appliance). After initial configuration, replication is fully automatic. Each Recovery Appliance in a replication topology manages its own protection and polling policies.

2.8.1 How a Downstream Recovery Appliance Processes Backups

To forward backups to a downstream Recovery Appliance, the upstream Recovery Appliance uses the same Recovery Appliance Backup Module that a protected database uses to send backups. The basic steps for processing backups are as follows:

1. A protected database uses its Recovery Appliance Backup Module to send backups to the Recovery Appliance.
2. The Recovery Appliance receives the backups and processes them as normal.
3. The upstream Recovery Appliance forwards the backups to the downstream Recovery Appliance.

Copyright © 2014, 2026, Oracle and/or its affiliates.

> **Note**
>
> The downstream Recovery Appliance does not know that it serves in the downstream role. The logic for receiving and processing backups on the downstream Recovery Appliance is independent of what occurs on the upstream Recovery Appliance.

> **Note**
>
> When real-time redo transport is enabled, incoming redo changes are not replicated in real time by Recovery Appliance. When an archived redo log backup is created, the Recovery Appliance automatically replicates this backup along with the data file backups.

4. As it receives backups from the upstream Recovery Appliance, the downstream Recovery Appliance updates its own metadata database.
5. The upstream Recovery Appliance requests metadata updates from the downstream metadata.

Periodically, the upstream Recovery Appliance requests metadata updates from the Recovery Appliances directly downstream from it. On receiving a metadata request, the downstream Recovery Appliance sends metadata updates to the upstream Recovery Appliance, which updates its own recovery catalog. In Recovery Appliance replication, this process is known as reconciling.

> **See Also**
>
> "Lifecycle of a Backup: Scenario"

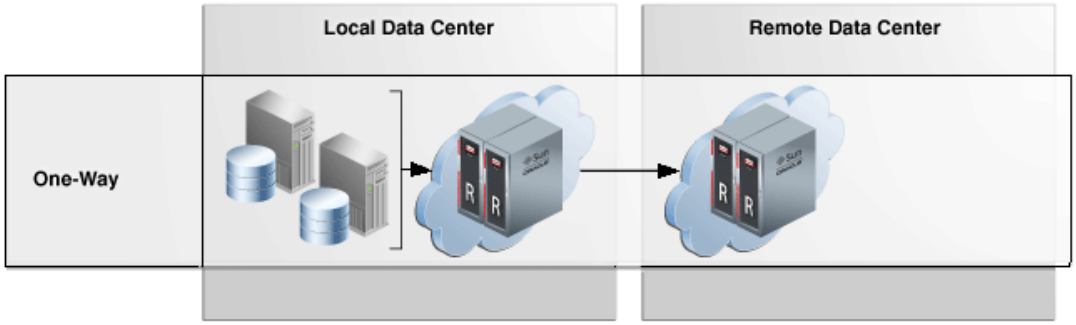
2.8.2 Replication Use Cases

Because a downstream Recovery Appliance processes backups independently from the upstream Recovery Appliance, a downstream Recovery Appliance can have completely different policies for every database whose backups it is storing. This design allows for a wide variety of use cases to be configured. In general, the use case for replication is to preserve database backup and recovery operations in the event of an outage or loss of the local recovery appliance. The replication topologies described below illustrate how this can be achieved.

- One-Way

  Backup data from local protected database (DB-a) flows to an upstream Recovery Appliance (RA-x), which forwards it to a downstream Recovery Appliance (RA-y).

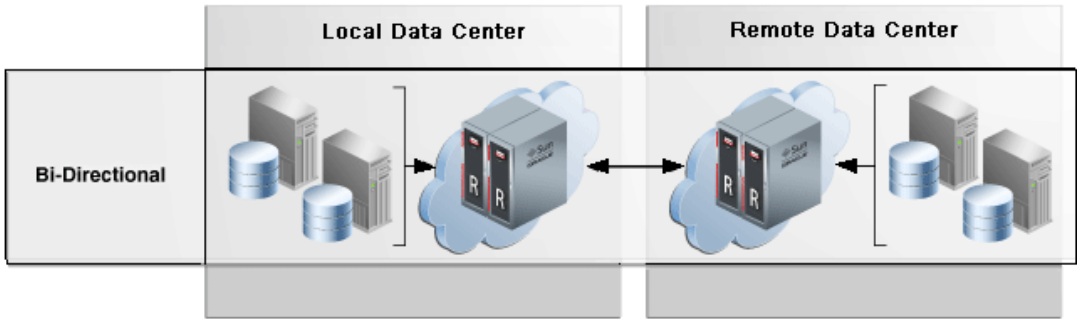
Copyright © 2014, 2026, Oracle and/or its affiliates.



- Bi-Directional

  Backup data from local protected databases (DB-a) flow to a local Recovery Appliance (RA-x), which then forwards them to a remote Recovery Appliance (RA-y). Conversely, backup data from remote database (DB-d) flow to a RA-y, which then forwards them to RA-x.

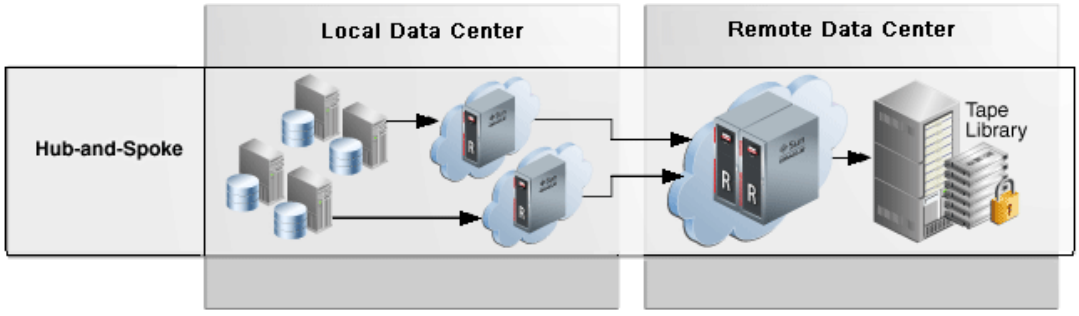
  Bi-directional replication is essentially one-way replication from RA-x to RA-y for a certain set of protected databases (DB-a,DB-b, DB-c), and also one-way replication the other direction from RA-y to RA-x for a different set of protected databases (DB-d, DB-e, DB-f).



  In this case, each Recovery Appliance plays both the upstream *and* downstream roles in the replication topology. Every Recovery Appliance serves as the primary backup location for one set of protected databases, and the secondary backup location for the other set. In this way, every Recovery Appliance is actively utilized while also providing disaster recovery services for the other Recovery Appliance.

- Hub-and-Spoke

  Backups flow from one set of databases to a local Recovery Appliance, and from a different set of databases to a different local Recovery Appliances. The local Recovery Appliances then forward these backups to a single remote Recovery Appliance, which archives the backups to tape.

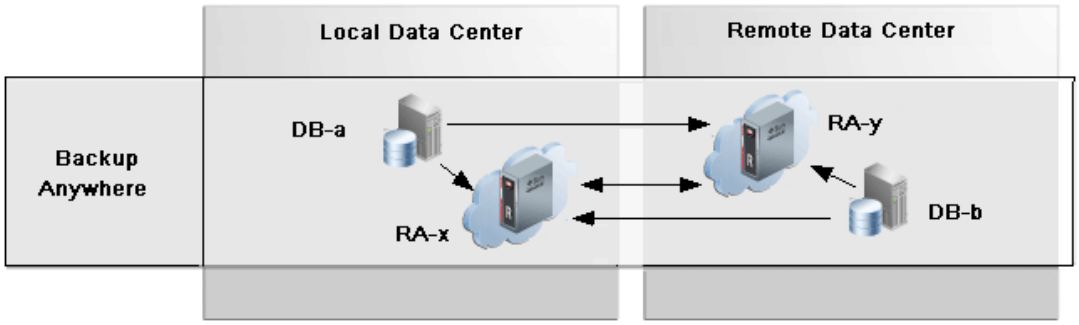


- Backup Anywhere

  Two Recovery Appliances RA-x and RA-y replicate between each other, and are the upstream and downstream of each other. Backups from databases DB-a and DB-b flow to

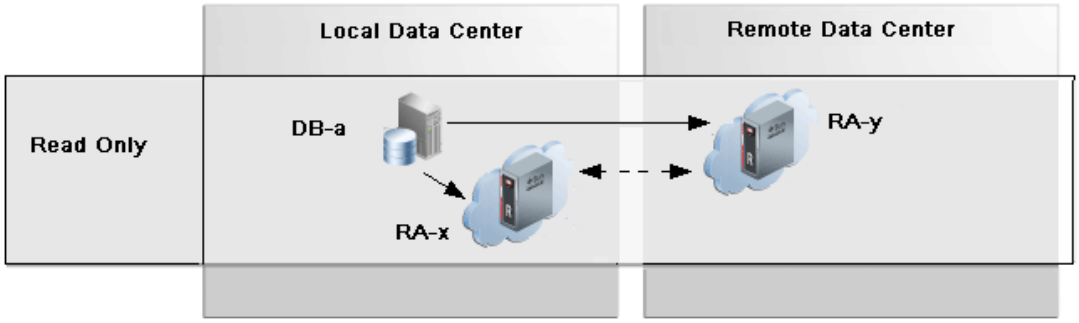
Copyright © 2014, 2026, Oracle and/or its affiliates.

their upstream Recovery Appliance, respectively RA-x and RA-y. Then RA-x and RA-y replicate these backups to their downstream Recovery Appliance, respectively RA-y and RA-x. Backups from databases DB-a and DB-b can be sent to either RA-x or RA-y with replication occurring to the Recovery Appliance which did not receive the backups directly from the DB.



- Read-Only Replication

  The `read_only` replication mode is useful when changing the destination Recovery Appliance for database backups from the original RA-x to a different RA-y while RA-x remains available to RA-y. This eliminates the need to replicate the old backups from RA-x to RA-y, which is time consuming and network intensive. If the original RA-x is available to RA-y, a replication server configuration in `read_only` mode can be added to the protection policy on RA-y which designates RA-x as the downstream. All backups for the databases in the policy that exist on the downstream are retrievable when and if needed by the upstream. After the old backups on the original RA-x have expired according to the protection policy, RA-y no longer needs the backups on RA-x which allows RA-x to be removed from the replication topology.



  A key use case for `read_only` replication is for introducing new Recovery Appliances into a replication configuration, so that the original Recovery Appliances can be decommissioned or so that protected database workload can be scaled out across additional Recovery Appliances. This allows for the graceful transfer of backup / restore between Recovery Appliances without unnecessary transmission and duplication of old backup files.

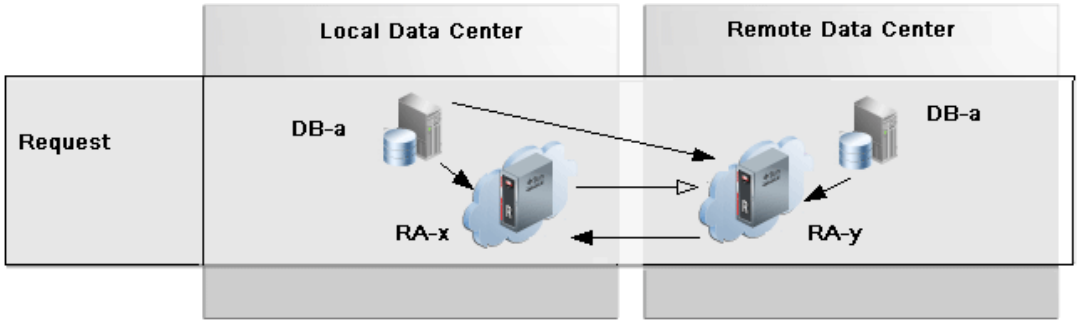
- Request Mode Replication

  In `request_only` mode replication, the upstream (RA-x) Recovery Appliance receives the primary database backups while the downstream (RA-y) receives the standby database backups, redo logs, and archive logs. When the upstream RA-x is offline for planned maintenance as one example, the primary database redo and archive logs are redirected to the downstream RA-y, where new archived log backups are created in order to preserve database recoverability. Redo and archived logs are the more critical data to retrieve from the upstream databases, because the database can hang if the local archived log directory fills up. Therefore, they are transmitted to RA-y so that they can be safely deleted by RMAN on DB-a.

Copyright © 2014, 2026, Oracle and/or its affiliates.

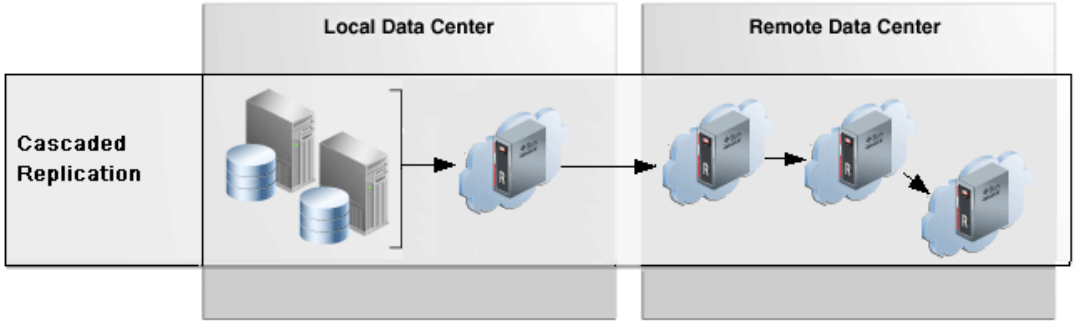
The command `RMAN CONFIGURE ARCHIVELOG DELETION POLICY BACKED UP 1 TIMES` establishes that when redo has been shipped and backed up on RA-y, then RA-x can safely delete the local archived logs to reclaim space. With this policy configured, the command `RMAN DELETE ARCHIVELOG` can also be used to safely reclaim local archived log space.

RA-y is not affected by RA-x outage and still receives level 1 backups from standby database as normal. Standby backups on RA-y can be used for primary database restores. When RA-x comes back online, all previously redirected backups are replicated from the downstream RA-y to the upstream RA-x to re-establish its full recoverability of its databases.



- Cascade Mode Replication

  Any of the preceding use cases could be adapted for cascaded replication, in which an upstream Recovery Appliance replicates to a downstream Recovery Appliance, which in turn replicates to another Recovery Appliance, creating a one-way chain of Recovery Appliances.



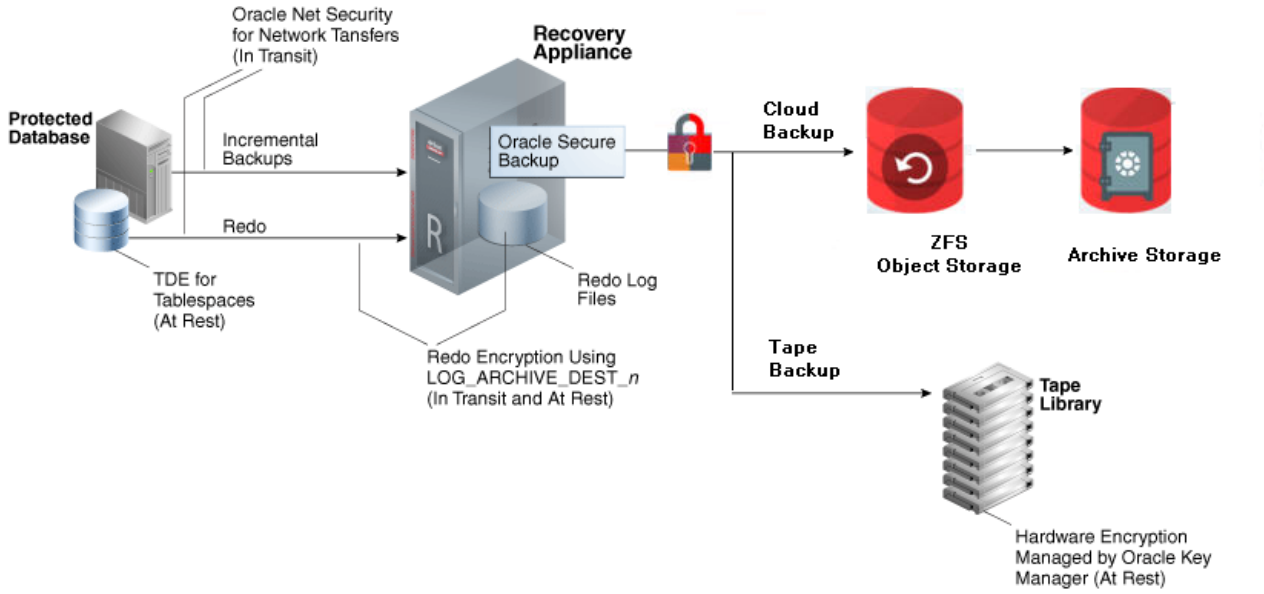
> **See Also**
>
> Replicating Backups with Recovery Appliance

2.9 Data Encryption Techniques

Various encryption options are available for backups and redo sent to the Recovery Appliance, as shown in Figure 2-9. The Recovery Appliance does not provide server-side encryption, which means that the appliance does not itself encrypt and decrypt data.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Figure 2-9 Data Encryption Techniques



The following types of encryption are supported:

- Transparent Data Encryption (TDE) on Production Database Tablespaces
- Redo Encryption Using LOG_ARCHIVE_DEST_n
- Tape Drive-Based Hardware Encryption

2.9.1 Transparent Data Encryption (TDE) on Production Database Tablespaces

Oracle recommends that you enable TDE on tablespaces in the database, and then take incremental backups as usual. TDE requires the Advanced Security Option. The benefits of TDE are as follows:

- TDE is transparent to applications.
- Backups of encrypted tablespaces, and redo describing changes to these tablespaces, are encrypted. The TDE-encrypted data blocks are secured on the protected database, Recovery Appliance storage, tape devices, and replicated appliances, and also when transferred through any network connections.
- TDE on the source database reduces overhead on downstream servers.
- TDE supports *space-efficient encrypted backups* by allowing the data file blocks to be decrypted before they are compressed and then re-encrypted.
- This technique supports an incremental-forever strategy and virtual full backups.

Copyright © 2014, 2026, Oracle and/or its affiliates.

> **Note**
>
> Space-efficient encrypted backups are enabled by specifying `RA_FORMAT=TRUE` in RMAN channel allocation commands with the Recovery Appliance and using `SET` or `CONFIGURE ENCRYPTION` command. See "Archival and Encrypted Backups" for more information

The following table shows the support for incremental forever when RMAN encryption and/or RMAN compression are used for the protected database backups:

Table 2-4 Support for Incremental Forever with RMAN Encryption and RMAN Compression

Data in the Database	No RMAN Encryption and No RMAN Compression	RMAN Encryption	RMAN Compression	RMAN Encryption and RMAN Compression
Not Encrypted	Yes; RMAN incremental backups.	No; incremental forever requires `RA_FORMAT=TRUE` in channel allocation and `SET ENCRYPTION ON`.	Yes; specify `RA_FORMAT=TRUE` in channel allocation.	Yes; specify `RA_FORMAT=TRUE` in channel allocation and `SET ENCRYPTION ON`.
TDE Tablespace Encryption	Yes; RMAN incremental backups.	Yes; specify `RA_FORMAT=TRUE` in channel allocation. TDE backups are always compressed and encrypted.	Yes; specify `RA_FORMAT=TRUE` in channel allocation. TDE backups are always compressed and encrypted.	Yes; specify `RA_FORMAT=TRUE` in channel allocation and `SET ENCRYPTION ON`.

> **See Also**
>
> - *Oracle Database Advanced Security Guide* to learn about TDE
> - *Oracle Database Backup and Recovery User's Guide* to learn about configuring backup encryption
> - *Oracle Database Backup and Recovery User's Guide* to learn about making compressed backups

2.9.2 Redo Encryption Using LOG_ARCHIVE_DEST_n

When enabled, the `ENCRYPTION` attribute of `LOG_ARCHIVE_DEST_`*n* encrypts redo both at rest on the Recovery Appliance and during the network transfer to the appliance. The basic process is as follows:

1. The protected database encrypts the redo in memory, using the private key contained in the Oracle Wallet on the protected database.
2. The protected database transfers the redo to the Recovery Appliance over the network.

Copyright © 2014, 2026, Oracle and/or its affiliates.

> **Note**
>
> If Oracle Net security is also enabled, then the redo is double encrypted during network transfer.

3. The Recovery Appliance writes the encrypted redo to archived redo log files, which exist in encrypted form only on the Recovery Appliance.

In a recovery scenario, RMAN restores and decrypts the encrypted redo log files on the protected database, using the encryption key stored in the Oracle wallet on the protected database host (not on the Recovery Appliance). RMAN never applies encrypted redo log files during media recovery.

> **See Also**
>
> - My Oracle Support Note Doc ID KB439308 for versions of Oracle AI Database that support encrypted redo
> - *Oracle Data Guard Concepts and Administration* to learn about redo encryption using `LOG_ARCHIVE_DEST_`*n*

2.9.3 Tape Drive-Based Hardware Encryption

The Recovery Appliance supports tape drive-based hardware encryption. In this case, the tape drive encrypts the data, not the software.

> **Note**
>
> Oracle Secure Backup can encrypt backup pieces before Recovery Appliance copies them to tape. However, Oracle does not recommend software-based encryption because of its possible negative effect on performance.

Copyright © 2014, 2026, Oracle and/or its affiliates.

> **Note**
>
> Oracle Secure Backup (OSB) 19.1 desupport was announced on May 1, 2026 with premier support end date on September 30, 2027. See My Oracle Support note PNEWS3035 for more information. Please refer to Lifetime Support Policy: Oracle Technology Products for updated support timeframes.
>
> Oracle recommends that database backups to cloud take advantage of Oracle Database Zero Data Loss Cloud Protect, which uses Oracle Zero Data Loss Autonomous Recovery Service in OCI. Cloud Protect offers an efficient incremental forever backup strategy, real-time transaction protection, logically air-gapped immutable backups, and fast, point-in-time recovery.
>
> For customers who require database backups to Amazon S3, Oracle Database Cloud Backup for Amazon S3 may be used. This offering is similar to Oracle Secure Backup Cloud Module in that it is fully integrated with Recovery Manager (RMAN) and enables you to back up your Oracle database to Amazon S3. You can exchange your Oracle Secure Backup license for an Oracle Database Cloud Backup for Amazon S3 license on a 1:1 basis. See Oracle Database Cloud Backup for Amazon S3 for more information.
>
> As with Oracle Secure Backup Cloud Module, Oracle Database Cloud Backup for Amazon S3 can be used free of charge on Oracle Zero Data Loss Recovery Appliance (ZDLRA) for the sole purpose of backing up data to and restoring data from S3-compatible storage devices that are in the same campus as ZDLRA.

For key management, Oracle recommends Oracle Key Manager, which centrally authorizes, secures, and manages all encryption keys. Oracle Key Manager does not consume CPU on the Recovery Appliance when encrypting and decrypting data.

> **See Also**
>
> *Oracle Secure Backup Administrator's Guide* to learn about hardware encryption

Copyright © 2014, 2026, Oracle and/or its affiliates.

Part I
Managing Recovery Appliance

Part I contains the following chapters:

- Recovery Appliance Workflow
- Getting Started with Cloud Control for Recovery Appliance
- Implementing Immutable Backups
- Managing Protection Policies with Recovery Appliance
- Configuring Recovery Appliance for Protected Database Access
- Copying Backups to Tape with Recovery Appliance
- Archiving Backups to Cloud
- Replicating Backups with Recovery Appliance
- Monitoring the Recovery Appliance
- Accessing Recovery Appliance Reports

Copyright © 2014, 2026, Oracle and/or its affiliates.

3

Recovery Appliance Workflow

This chapter explains the basic workflow for managing a Zero Data Loss Recovery Appliance environment. Where appropriate, this chapter refers to ZDLRA Protected Database Configuration Guide . The chapter contains the following topics:

- Separation of Duties in Recovery Appliance Administration
- Prerequisites for Recovery Appliance Administration
- Tools for Recovery Appliance Administration
- Planning for Recovery Appliance
- Setup and Configuration for Recovery Appliance
- Maintenance Tasks for Recovery Appliance

3.1 Separation of Duties in Recovery Appliance Administration

A typical Recovery Appliance environment includes personnel with the following roles:

- Cloud Control administrator

  The application administrator with this role administers Oracle Enterprise Manager Cloud Control (Cloud Control). Duties may include:

  - Discovering targets, including the Recovery Appliance
  - Managing one or more protected databases
  - Managing one or more Recovery Appliances
- Recovery Appliance administrator

  This administrator manages Recovery Appliance. Typical duties include:

  - Creating the protection policies
  - Assigning protected databases to protection policies
  - Managing space on Recovery Appliance
  - Configuring tape and replication operations
  - Creating the Recovery Appliance user accounts that own virtual private catalogs
  - Monitoring Recovery Appliance, and generating reports
- Protected database administrator

  This administrator is responsible for configuring backups to the Recovery Appliance using the virtual private catalog account assigned by the Recovery Appliance administrator.

Copyright © 2014, 2026, Oracle and/or its affiliates.

See Also

- "User Accounts in the Recovery Appliance Environment"
- *Oracle Database Security Guide* to learn how to create database user accounts

3.2 Prerequisites for Recovery Appliance Administration

You must work with the Oracle field engineers to install and set up Recovery Appliance.

3.2.1 Tools for Recovery Appliance Administration

Use the following tools to complete administrative tasks for Recovery Appliance:

- Cloud Control

  Cloud Control is a system management tool with a graphical user interface that enables you to manage and monitor Recovery Appliance and its protected databases. This is the preferred UI for Recovery Appliance tasks.

- SQL*Plus

  SQL*Plus is a command-line tool that enables you to run `DBMS_RA` program units, and query recovery catalog views. You use SQL statements and Oracle-supplied PL/SQL packages to complete these tasks in SQL*Plus.

  See *SQL*Plus User's Guide and Reference*.

3.2.2 Planning for Recovery Appliance

You must complete the following general tasks:

- Task 1: Group protected databases into tiers
- Task 2: Determine the recovery requirements for each database tier
- Task 3: Determine the recovery requirements for each protected database
- Task 4: Determine access requirements for Recovery Appliance
- Task 5: Create a backup migration plan to Recovery Appliance
- Task 6: Review Cloud Control reporting and monitoring tools

Task 1: Group protected databases into tiers

Group databases based on their recovery requirements. By default, Recovery Appliance includes the protection policies Platinum, Gold, Silver, and Bronze. Each policy corresponds to a level of protection. For example, Gold provides databases in this tier with real-time redo transport protection, whereas Bronze does not.

Task 2: Determine the recovery requirements for each database tier

For each database tier, make decisions about the following:

- The maximum amount of time for potential data loss exposure
- The disk recovery window goal
- The recovery window for tape

Copyright © 2014, 2026, Oracle and/or its affiliates.

- The schedule for database tiers that back up to tape, and any tape vaulting or encryption requirements
- Whether to configure Recovery Appliance replication
- Directories for backup polling, if you intend to enable a backup polling policy
- Whether existing recovery catalogs will be imported into the Recovery Appliance catalog
- Whether to enable the guaranteed copy feature, which requires that backups on Recovery Appliance be copied to tape or replicated before being considered for deletion to reclaim space
- The maximum retention time of backups on disk

> **See Also**
>
> "About Protection Policies"

Task 3: Determine the recovery requirements for each protected database

For example, perform the following tasks:

- Calculate the reserved space, which is based on the protected database size, change rate, and recovery window goal.

  The protection policy can make use of `autotune_reserved_space` parameter if compliance features not required. When enabled, the Recovery Appliance automatically defines and updates the `reserved_space` (the minimum allocated per database) based on computed `recovery_window_space` to meet recovery window goal, up to the total available free space. This is handled for all databases associated with this policy.

  For compliance backups, `reserved_space` is a hard limit allocated for a given database, so `autotune_reserved_space` does not apply.

  When database is first created or resumed without explicit `reserved_space` set and `autotune_reserved_space` enabled, it has the state `PROVISIONAL`. Once a resync and an initial backup provided, it transitions into an `ACTIVE` or enabled state.
- Decide whether to implement real-time redo transport

See *Zero Data Loss Recovery Appliance Protected Database Configuration Guide* for additional planning considerations for protected databases.

Task 4: Determine access requirements for Recovery Appliance

Decide which persons have access to the Recovery Appliance in the data center. For example, database administrators, storage administrators, system administrators, and backup administrators may have different access requirements. In some data centers, a single person may play all roles.

Task 5: Create a backup migration plan to Recovery Appliance

In this stage, decide how your legacy RMAN backups fit into your Recovery Appliance backup strategy. After setting up Recovery Appliance, you may choose either of the following strategies:

- Continue to run old backups to disk and tape concurrently with new backups to Recovery Appliance for a specified time, until you are ready to back up to Recovery Appliance exclusively.

Copyright © 2014, 2026, Oracle and/or its affiliates.

- Back up protected databases exclusively to Recovery Appliance, and then manage legacy backups on legacy media separately.

In either case, to simplify overall catalog management, Oracle recommends that you first import legacy RMAN recovery catalogs into the Recovery Appliance catalog.

> **See Also**
>
> *Zero Data Loss Recovery Appliance Protected Database Configuration Guide* to learn how to import metadata into the Recovery Appliance catalog

Task 6: Review Cloud Control reporting and monitoring tools

Cloud Control is the preferred interface for Recovery Appliance. Before configuring Recovery Appliance, become familiar with the main Cloud Control pages, as described in Getting Started with Cloud Control for Recovery Appliance . Database administrators can also review backup-related pages such as Backup Settings, Schedule Backup, and Backup Reports.

> **See Also**
>
> - Monitoring the Recovery Appliance
> - Accessing Recovery Appliance Reports

3.3 Setup and Configuration for Recovery Appliance

You must complete the following general tasks:

- Task 1: Create Cloud Control user accounts
- Task 2: Create a protection policy for each database tier
- Task 3: Configure access on Recovery Appliance for protected databases
- Task 4: Configure protected databases (for DBAs)
- Task 5: Migrate legacy backups to Recovery Appliance (for DBAs)
- Task 6: Create copy-to-tape schedules to meet recovery requirements
- Task 7: Configure Recovery Appliance replication

Task 1: Create Cloud Control user accounts

As explained in "Separation of Duties in Recovery Appliance Administration", a Recovery Appliance environment may require multiple administrative accounts. In this step, create the Cloud Control user accounts necessary for your environment.

> **Note**
>
> These are application-level user accounts, not database user accounts.

Copyright © 2014, 2026, Oracle and/or its affiliates.

> **See Also**
>
> Cloud Control help to learn how to create Enterprise Manager user accounts

Task 2: Create a protection policy for each database tier

For each tier of protected databases, create a separate protection policy. "Basic Tasks for Managing Protection Policies" describes these tasks.

1. Optionally, if your Recovery Appliance has access to a backup polling location, then create a backup polling policy.

   > **Note**
   >
   > If you are using Cloud Control, then this step is included in the protection policy configuration. When using `DBMS_RA`, you must run a separate procedure (`CREATE_POLLING_POLICY`).

   "Creating a Backup Polling Policy (Command-Line Only)" describes this task.

2. Create a protection policy for a specific database tier.

   "Creating a Protection Policy" describes this task.

Task 3: Configure access on Recovery Appliance for protected databases

Create a virtual private catalog owner in the Recovery Appliance metadata database, add protected database metadata, and grant the catalog owner access to protected databases. Perform all of these steps on the Recovery Appliance, as explained in "Basic Tasks for Configuring Protected Database Access".

Task 4: Configure protected databases (for DBAs)

Protected database administrators perform this task, which does not involve running `DBMS_RA` procedures on Recovery Appliance. Client-side configuration includes the following subtasks:

1. Configuring backup and recovery settings, including real-time redo transport
2. Enabling access to the Recovery Appliance, which involves installing the Recovery Appliance Backup Module and authenticating the Recovery Appliance User Account
3. Testing backup and restore operations

See *Zero Data Loss Recovery Appliance Protected Database Configuration Guide* to learn how to configure protected databases.

Task 5: Migrate legacy backups to Recovery Appliance (for DBAs)

DBAs for protected databases perform this task, which does not involve running `DBMS_RA` procedures on Recovery Appliance. Migration includes importing legacy recovery catalogs into the Recovery Appliance catalog, and enabling the Recovery Appliance to access physical backups on disk or tape.

See *Zero Data Loss Recovery Appliance Protected Database Configuration Guide* to learn how to migrate legacy backups.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Task 6: Create copy-to-tape schedules to meet recovery requirements

If you employ tape devices in your environment, then you must create SBT attribute sets, schedule tape jobs, monitor tape backup status, and so on. You perform all of these steps on the Recovery Appliance, as explained in "Basic Tasks for Copying Backups to Tape with Recovery Appliance".

Task 7: Configure Recovery Appliance replication

This task involves configuring both the upstream Recovery Appliance and the downstream Recovery Appliance, and performing some steps on the protected database hosts.

3.4 Maintenance Tasks for Recovery Appliance

Typically, you must perform the following tasks:

- Task 1: Monitor activity on Recovery Appliance
- Task 2: Monitor backup jobs (for DBA)
- Task 3: Generate and review reports on Recovery Appliance
- Task 4: Restart the Recovery Appliance

Task 1: Monitor activity on Recovery Appliance

Using Cloud Control, monitor Recovery Appliance to ensure that business requirements are being met. For example, do the following:

- Review any alerts or warnings
- Verify that available space can meet all recovery windows
- Verify that backup throughput meets performance requirements

See "Basic Tasks for Monitoring the Recovery Appliance".

Task 2: Monitor backup jobs (for DBA)

Protected database administrators must periodically monitor backup job reports for errors.

See *Zero Data Loss Recovery Appliance Protected Database Configuration Guide*.

Task 3: Generate and review reports on Recovery Appliance

Using Cloud Control, generate and review BI Publisher reports for storage usage and capacity planning.

See "Basic Tasks for Accessing Recovery Appliance Reports".

Task 4: Restart the Recovery Appliance

If necessary, shut down and start up the Recovery Appliance using operating system utilities and `DBMS_RA` procedures. See *Zero Data Loss Recovery Appliance Owner's Guide* to learn how to restart the Recovery Appliance.

Copyright © 2014, 2026, Oracle and/or its affiliates.

4

Getting Started with Cloud Control for Recovery Appliance

This chapter explains how to access the principal pages in Oracle Enterprise Manager Cloud Control (Cloud Control) for Recovery Appliance, and contains the following sections:

- Displaying All Recovery Appliances in the Enterprise
- Accessing the Recovery Appliance Home Page
- Accessing the Recovery Appliance Storage Locations Page

4.1 Displaying All Recovery Appliances in the Enterprise

Cloud Control lists every Recovery Appliance in the enterprise. From this page, you can access the individual home page of any Recovery Appliance.

To display all Recovery Appliances in the enterprise:

1. On the Cloud Control Login page, enter your `SYSMAN` user name and password.

   The Welcome to Enterprise Manager Cloud Control page appears.

   > **See Also**
   >
   > "User Accounts in the Recovery Appliance Environment" for more information on user accounts in the Recovery Appliance environment

2. Select **Targets**, and then **Recovery Appliances**.

   The Recovery Appliances page provides an overview of all Recovery Appliances in your environment. You can click the links in some columns (for example, Member Status or Incidents and Events) to go to pages with more information.

3. From the table of Recovery Appliances, select the one you are interested in.

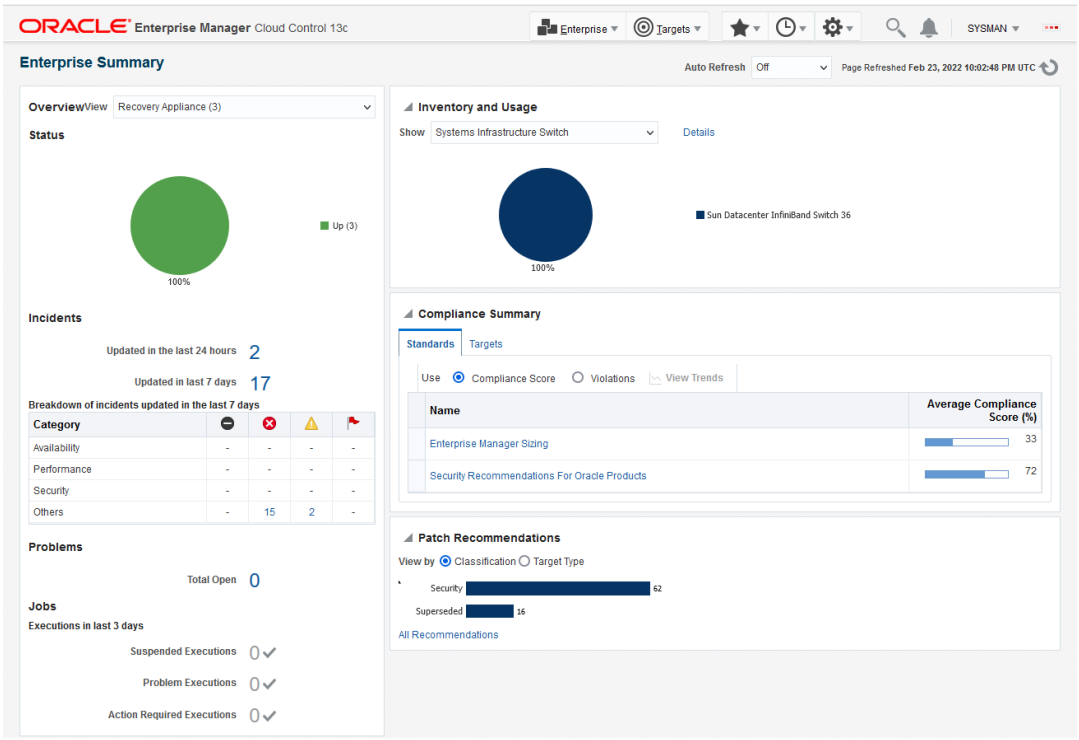
4. To see a pie chart showing the availability of the Recovery Appliances, select **Enterprise**, and then **Summary**.

   The Enterprise Summary page appears.

5. In the Overview section, in the **View** menu, select **Recovery Appliance**.

   The pie chart in the Status section indicates the percentage of Recovery Appliances that are available.

Copyright © 2014, 2026, Oracle and/or its affiliates.



4.2 Accessing the Recovery Appliance Home Page

The Recovery Appliance Home page is a command center that centralizes management of the Recovery Appliance environment. From this page, you can manage Recovery Appliance storage and performance, and view recent activity and issues that may need attention.

To access the Recovery Appliance Home page:

1. On the Cloud Control Login page, enter your `SYSMAN` user name and password.

> **See Also**
>
> "User Accounts in the Recovery Appliance Environment" for more information on user accounts in the Recovery Appliance environment

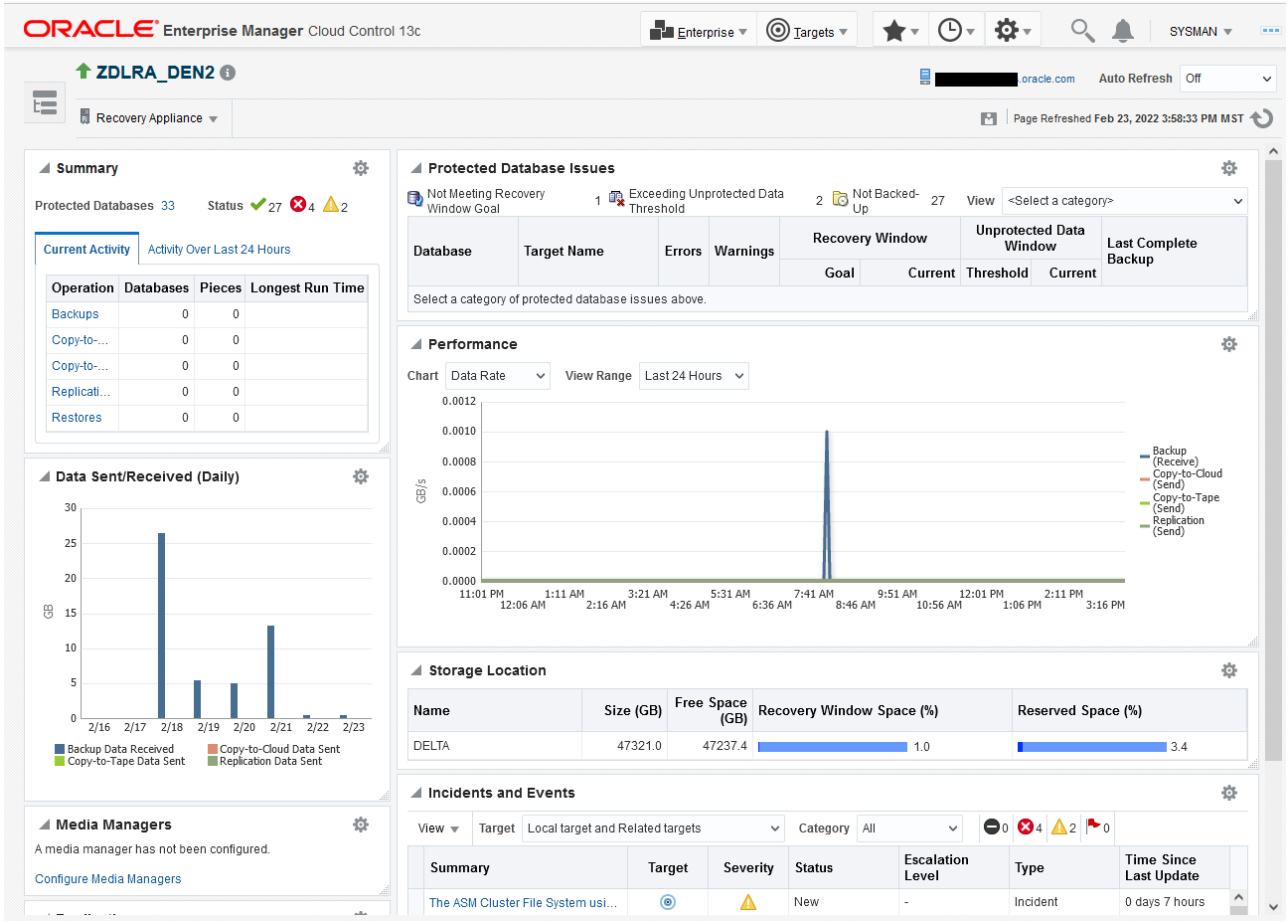
2. From any Cloud Control page, select **Targets**, and then **Recovery Appliances**.

   The Recovery Appliances page appears.

3. In the Name column, click the name of a Recovery Appliance.

   The Home page for the selected Recovery Appliance page appears. The following graphic shows part of a sample Home page:

Copyright © 2014, 2026, Oracle and/or its affiliates.



From this page you can see a snapshot of the entire Recovery Appliance, and also click links to obtain more information about a particular area.

4. Optionally, to access the main menu, click **Recovery Appliance**.

   The menu appears with many options, and many of those having additional fly-out menu options.

   - Home
   - Monitoring
   - Diagnostics
   - Control
   - Job Activity
   - Members
   - Reports
   - Protected Databases
   - Protection Policies
   - Replication
   - Archival Backups
   - Copy-to-Media Job Templates

Copyright © 2014, 2026, Oracle and/or its affiliates.

- Media Managers
- Storage Location
- Configuration
- Compliance
- Target Setup
- Target Sitemap
- Target Information

From the preceding menu you can go to all pages relating to management, monitoring, and reporting for this Recovery Appliance.

The Recovery Appliance Home page is divided into the following sections:

- **Summary**

  This section shows the number of protected databases, and summarizes their health status, current activity, and activity within the last 24 hours. For more information, click the links in the Operation column: Backup, Copy-to-Tape, Replication, and Restore.

- **Protected Database Issues**

  This section highlights any issues relating to backup and recovery status for protected databases. The View menu filters the information on key categories.

- **Data Sent/Received (Daily)**

  This section displays daily throughput over the past week.

- **Performance**

  This section charts performance statistics for Data Rate and Queued Data. The statistics are filterable by day, week, or month.

- **Media Managers**

  This section displays the configured media manager for copy-to-tape operations.

- **Storage Locations**

  This section summarizes total available space and usage by indicating how much has been consumed to meet the disk recovery window goal for all databases, and what percentage of total space is reserved space for databases backing up to the specified storage location (see "How Recovery Appliance Manages Storage Space").

- **Replication**

  This section lists the downstream Recovery Appliances to which this Recovery Appliance is replicating, and also the upstream Recovery Appliances from which this Recovery Appliance is receiving backups (see "About Replication Partners and Users Accounts").

- **Incidents and Events**

  This section summarizes all warnings or alerts that have been generated by Cloud Control monitoring of all targets associated with the Recovery Appliance. From this section, drill down for further detail on the issues.

4.3 Accessing the Recovery Appliance Storage Locations Page

The Storage Locations page expands on the information provided in the Storage Locations section on the Recovery Appliance Home page. This page provides the following storage-related information:

Copyright © 2014, 2026, Oracle and/or its affiliates.

- Oracle ASM disk groups in the Recovery Appliance storage location
- Number of protection policies using the storage location
- Total space (in GB) needed to meet disk recovery window goals for all databases protected by this Recovery Appliance
- Total reserved space for all databases protected by this Recovery Appliance

Besides using this page to monitor existing storage, you can use this page add storage to an existing storage location, or to create a new storage location.

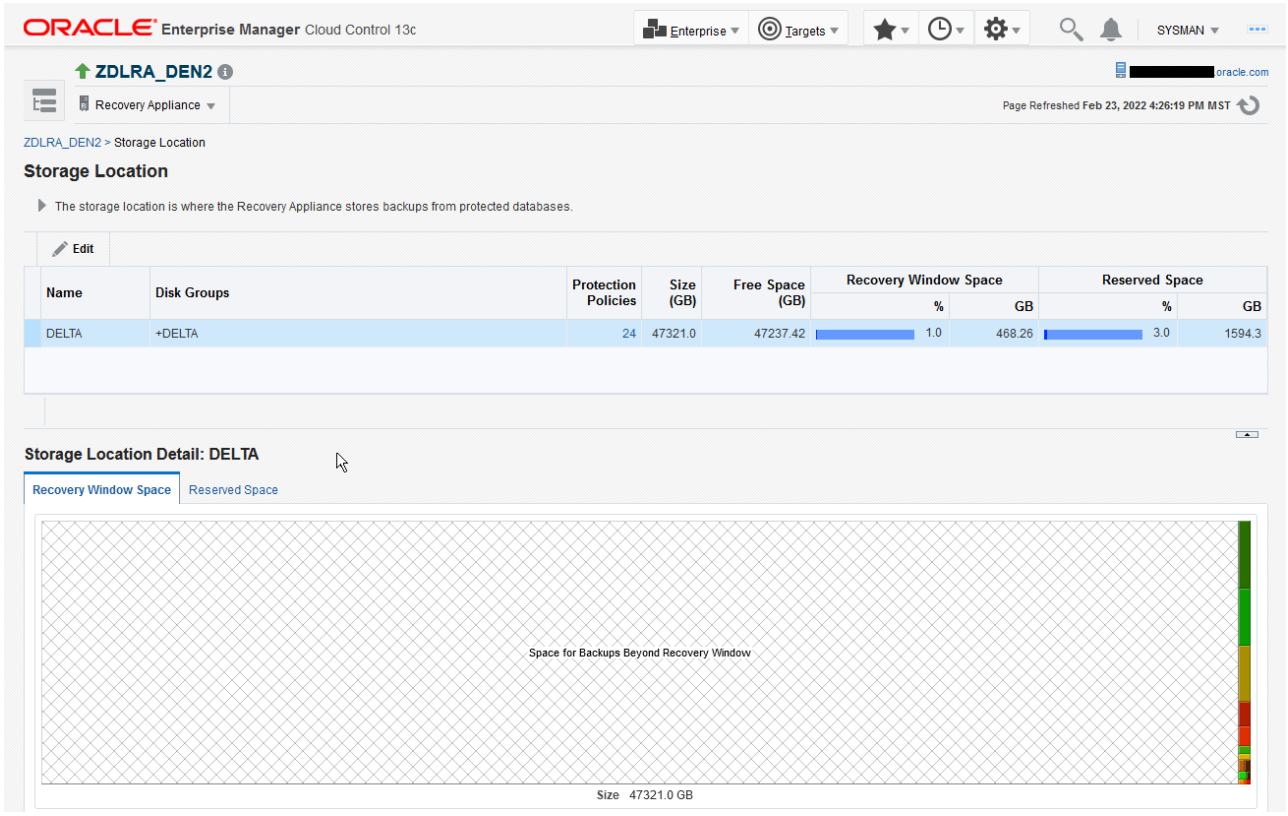
To access the Storage Locations page:

1. Access the Recovery Appliance Home page, as described in "Accessing the Recovery Appliance Home Page".
2. From the **Recovery Appliance** menu, select **Storage Location**.

   The Recovery Appliance login page appears when you first log in to the Recovery Appliance pages, or when the browser has been inactive for an extended time.
3. If prompted, enter your login credentials, and then click **Login**.

   The Storage Locations page appears, as shown in Figure 4-1.

Figure 4-1 Storage Locations Page



This page provides a useful graphical representation of how much of the storage location is reserved and unreserved. In the preceding sample page, the `DELTA` storage location, which is the default, is the only storage location configured.

Copyright © 2014, 2026, Oracle and/or its affiliates.

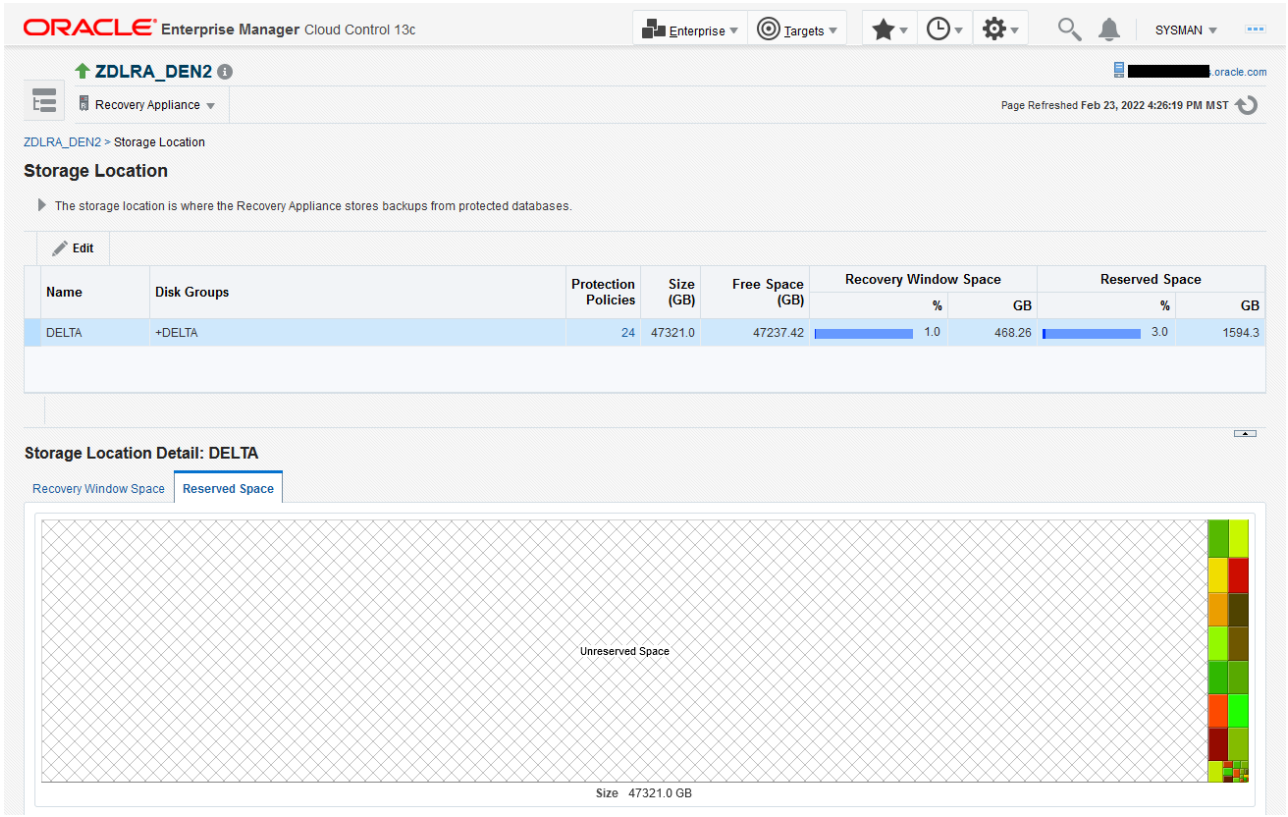
4. In the Storage Location Detail: DELTA section, click **Recovery Window Space** (if it is not already selected).

   The Recovery Window Space subpage appears.

   In the sample shown in Figure 4-1, the size of the `DELTA` storage location is 104491.8 GB and the total size of the needed recovery window space is 50813.0 GB. When the cursor hovers over the name of a protected database, as for the `CUSTOMER` database, a message indicates the amount of storage space needed by this database to meet its recovery window, and the percentage of the total storage space required.

5. In the Storage Location Detail: DELTA section, click **Reserved Space**.

   The Reserved Space subpage appears.



In the preceding graphic, the total size of the reserved space is 94064.0 GB. When the cursor hovers over the name of a protected database, as for the `CUSTOMER` database, a message indicates the amount of reserved space needed by this database, and the percentage of the total storage space required.

Copyright © 2014, 2026, Oracle and/or its affiliates.

5

Securing the Operations of the Recovery Appliance

The following steps harden the Recovery Appliance by reducing exposure to powerful users, like `root` and `rasys` and allowing improved auditing of maintenance actions. Although this procedure is optional for many installations and applications, establishing and using secure users is required for operations to be compliant with various regulatory mandates.

For purposes of example, the sample commands have three fictive users: `bob`, `sue`, and `jim`.

1. **Create named users and assign them `db_user` with user type `admin` with administration rights.**

   The `db_user` user type `admin` replaces the usage of `rasys` for configuration and day-to-day Recovery Appliance management operations. This account can issue certain SQLPlus commands within its assigned privileges.

   ```
   racli add db_user --user_type=admin --user_name=bob
   racli add db_user --user_type=admin --user_name=sue
   ```

   In this example, `bob` and `sue` are given `--user_type=admin` for administration rights.

   > **Note**
   >
   > The `db_user` user type `admin` has limits of privileges, and cannot be used as `sysdba` in SQLPlus.

2. **Create `ssh` users for the Recovery Appliance.**

   The `admin_user` account is a role for new named users who manage the Recovery Appliance from an operation's perspective. It permits operating system level operations on the Recovery Appliance that previously required `root` access, however `admin_user` is not `root`.

   ```
   racli add admin_user --user_name=bob
   racli add admin_user --user_name=jim
   racli add admin_user --user_name=sue
   ```

   In this example, `bob`, `sue` and `jim` are given `admin_user` with administration rights.

3. **Disable `ssh` access for `root` and `oracle`.**

   ```
   racli disable ssh
   ```

4. **Disable `root` access for `root`, `oracle`, and `raadmin`.**

   ```
   racli disable root_access
   ```


Copyright © 2014, 2026, Oracle and/or its affiliates.

5. **Disable `rasys` access.**

   > **Note**
   >
   > Make sure that you have the `db_user` user type `admin` accounts and `admin_user` accounts before disabling `rasys` access.

   ```
   racli disable rasys_user
   ```

6. **Disable sys remote access.**

   ```
   racli disable sys_remote_access
   ```

7. **Validate the time service.**

   Refer to Changing the CHRONY Servers.

8. **Validate that the Recovery Appliance is in compliance.**

   ```
   racli run check --check_name=check_ra_compliance
   ```

   The above should return `TRUE`. The `check_ra_compliance` validates:

   - `ssh` access for `root` and `oracle` is disabled on all nodes.
   - `rasys` access is disabled.
   - `sys` remote access is disabled.
   - Time service is enabled.
   - Two or more `admin_users` for the Recovery Appliance have been established.
   - Two or more `db_users` who are `admin` have been established.

   If any of the above items are not completed, `check_ra_compliance` fails, because one or more security gaps still exist on the Recovery Appliance.

At the completion of the above steps:

- The initial set of administrative users have been configured.
- An audit trail of actions by administrative users is now possible.
- Various commands are restricted to users with the proper permissions.
- Certain commands are restricted to *quorum* operations requiring approval of others to finally be run.

5.1 Remote Handling of Recovery Appliance System Logs

As part of efficient management of the Recovery Appliance, it can be beneficial to export the system log files automatically to one or more remote servers for status monitoring and review.

You can configure which Recovery Appliance log files are sent, such as:

- `/var/log/audit/audit.log`
- `/var/log/messages`

Copyright © 2014, 2026, Oracle and/or its affiliates.

- /var/log/oracle/deploy/dbmcli.lst.root.0
- /var/log/aide/aide.log
- /etc/passwd
- /var/log/yum.log
- /var/log/clamav/clamscan.log
- /var/log/secure
- /opt/oracle.RecoveryAppliance/log/ra_export.log
- /opt/oracle.RecoveryAppliance/log/em_backup.log
- /opt/oracle.RecoveryAppliance/log/ra_fs_cleanup.log
- /opt/oracle.RecoveryAppliance/log/emctl.log
- /opt/oracle.RecoveryAppliance/log/racli_update_parameter.log
- /opt/oracle.RecoveryAppliance/log/racli_alter_parameter.log
- /opt/oracle.RecoveryAppliance/log/racli_list_parameter.log

To Create a Configuration File for a Remote Receiver

The command `racli add remote_syslog` creates a configuration file in `/etc/rsyslog.d/` from the arguments passed in:

```
racli add remote_syslog --dest=<desturl> --port=<destPort> --
config_name=<yourConfig>
```

- `--dest=<desturl>` defines the IP address of the (remote) destination to receive this Recovery Appliance's system logs.
- `--port=<destPort>` defines the port on the (remote) destination to receive this Recovery Appliance's system logs.
- `--config_name=<yourConfig>` defines a meaningful name to the organization, like `fleet01_remote_central`.

```
racli add remote_syslog --dest=100.104.102.184 --port=514 --
config_name=fleet1_test02:

Created log /opt/oracle.RecoveryAppliance/log/racli_add_remote_syslog.log
Mon Apr 11 09:17:41 2022: Start: Configure Sys Log to 100.104.102.184
Mon Apr 11 09:17:41 2022:   Start: On Local Node zdlra10adm01
Mon Apr 11 09:17:41 2022:       Start: Restart rsyslog
Mon Apr 11 09:17:41 2022:       End: Restart rsyslog
Mon Apr 11 09:17:41 2022:   End: On Local Node zdlra10adm01
Mon Apr 11 09:17:42 2022:   Start: On Remote Node zdlra10adm02
Mon Apr 11 09:17:43 2022:   End: On Remote Node zdlra10adm02
Mon Apr 11 09:17:43 2022: End: Configure Sys Log to 100.104.102.184
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

To View the Remote Receivers

The command racli list remote_syslog lists all the configuration files, or a specific one, from the `/etc/rsyslog.d/` directory.

```
racli list remote_syslog --config_name=fleet1_test01:

syslog_fleet1_test01:
    NAME = fleet1_test01
    CONFIG_FILE = /etc/rsyslog.d/fleet1_test01.conf
```

To Remove the Remote Receivers

The command racli remove remote_syslog removes a named configuration file from the `/etc/rsyslog.d/` directory.

```
racli remove remote_syslog --config_name='fleet1_test01'

Created log /opt/oracle.RecoveryAppliance/log/
racli_remove_remote_syslog.logMon Apr
1109:17:582022: Start: Remove Sys Log
fleet1_test01
Mon Apr 1109:17:582022:   Start: On Local Node zdlra10adm01
Mon Apr 1109:17:582022:       Removed: Sys Log fleet1_test01.conf
Mon Apr 1109:17:582022:       Removed: Metadata of syslog_fleet1_test01
Mon Apr 1109:17:582022:       Start: Restart rsyslog
Mon Apr 1109:17:582022:       End: Restart rsyslog
Mon Apr 1109:17:582022:   End: On Local Node zdlra10adm01
Mon Apr 1109:17:582022:   Start: On Remote Node zdlra10adm02
Mon Apr 1109:18:002022:   End: On Remote Node zdlra10adm02
Mon Apr 1109:18:002022: End: Remove Sys Log fleet1_test01
```

To Configure the Syslog Server or Fleet Manager

The external and separate syslog or fleet server needs to be configured to receive the Recovery Appliance log files.

- Each `config` file can accept one (1) destination only.
- The location of the `config` file is: `/etc/rsyslog.d/`
- Location set for logs: `/var/odo/hostsyslogs/`
- Naming convention on the log files: `%PROGRAMNAME%_%HOSTNAME%_%$YEAR%-%$MONTH%-%$DAY%-%$HOUR%.log`

Example of .conf file under /etc/rsyslog.d

```
##########REMOTE SYSLOG#################

  $ModLoad imfile

    ####################
    $InputFilePollInterval 180
    $InputFileName /var/log/aide/aide.log
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
$InputFileTag aide:
$InputFileStateFile stat-aide
$InputFileSeverity Info
$InputRunFileMonitor

if $programname == 'aide' then @@100.104.102.184:514
if $programname == 'aide' then stop

####################
$InputFilePollInterval 180
$InputFileName /var/log/audit/audit.log
$InputFileTag audit:
$InputFileStateFile stat-audit
$InputFileSeverity Info
$InputRunFileMonitor

if $programname == 'audit' then @@100.104.102.184:514
if $programname == 'audit' then stop

####################
$InputFilePollInterval 180
$InputFileName /var/log/clamav/clamscan.log
$InputFileTag clamav:
$InputFileStateFile stat-clamav
$InputFileSeverity Info
$InputRunFileMonitor

if $programname == 'clamav' then @@100.104.102.184:514
if $programname == 'clamav' then stop

####################
$InputFilePollInterval 180
$InputFileName /var/log/oracle/deploy/dbmcli.lst.root.0
$InputFileTag dbmcli:
$InputFileStateFile stat-dbmcli
$InputFileSeverity Info
$InputRunFileMonitor

if $programname == 'dbmcli' then @@100.104.102.184:514
if $programname == 'dbmcli' then stop

####################
$InputFilePollInterval 180
$InputFileName /opt/oracle.RecoveryAppliance/log/em_backup.log
$InputFileTag em-backup:
$InputFileStateFile stat-em-backup
$InputFileSeverity Info
$InputRunFileMonitor

if $programname == 'em-backup' then @@100.104.102.184:514
if $programname == 'em-backup' then stop

####################
$InputFilePollInterval 180
$InputFileName /opt/oracle.RecoveryAppliance/log/emctl.log
$InputFileTag emctl:
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
$InputFileStateFile stat-emctl
$InputFileSeverity Info
$InputRunFileMonitor

if $programname == 'emctl' then @@100.104.102.184:514
if $programname == 'emctl' then stop

####################
$InputFilePollInterval 180
$InputFileName /var/log/messages
$InputFileTag messages:
$InputFileStateFile stat-messages
$InputFileSeverity Info
$InputRunFileMonitor

if $programname == 'messages' then @@100.104.102.184:514
if $programname == 'messages' then stop

####################
$InputFilePollInterval 180
$InputFileName /etc/passwd
$InputFileTag passwd:
$InputFileStateFile stat-passwd
$InputFileSeverity Info
$InputRunFileMonitor

if $programname == 'passwd' then @@100.104.102.184:514
if $programname == 'passwd' then stop

####################
$InputFilePollInterval 180
$InputFileName /opt/oracle.RecoveryAppliance/log/ra_export.log
$InputFileTag ra-export:
$InputFileStateFile stat-ra-export
$InputFileSeverity Info
$InputRunFileMonitor

if $programname == 'ra-export' then @@100.104.102.184:514
if $programname == 'ra-export' then stop

####################
$InputFilePollInterval 180
$InputFileName /opt/oracle.RecoveryAppliance/log/ra_fs_cleanup.log
$InputFileTag ra-fs-cleanup:
$InputFileStateFile stat-ra-fs-cleanup
$InputFileSeverity Info
$InputRunFileMonitor

if $programname == 'ra-fs-cleanup' then @@100.104.102.184:514
if $programname == 'ra-fs-cleanup' then stop

####################
$InputFilePollInterval 180
$InputFileName /opt/oracle.RecoveryAppliance/log/racli_alter_parameter.log
$InputFileTag racli-alter-parameter:
$InputFileStateFile stat-racli-alter-parameter
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
    $InputFileSeverity Info
    $InputRunFileMonitor

    if $programname == 'racli-alter-parameter' then @@100.104.102.184:514
    if $programname == 'racli-alter-parameter' then stop

    ####################
    $InputFilePollInterval 180
    $InputFileName /opt/oracle.RecoveryAppliance/log/racli_list_parameter.log
    $InputFileTag racli-list-parameter:
    $InputFileStateFile stat-racli-list-parameter
    $InputFileSeverity Info
    $InputRunFileMonitor

    if $programname == 'racli-list-parameter' then @@100.104.102.184:514
    if $programname == 'racli-list-parameter' then stop

    ####################
    $InputFilePollInterval 180
    $InputFileName /opt/oracle.RecoveryAppliance/log/
racli_update_parameter.log
    $InputFileTag racli-update-parameter:
    $InputFileStateFile stat-racli-update-parameter
    $InputFileSeverity Info
    $InputRunFileMonitor

    if $programname == 'racli-update-parameter' then @@100.104.102.184:514
    if $programname == 'racli-update-parameter' then stop

    ####################
    $InputFilePollInterval 180
    $InputFileName /var/log/secure
    $InputFileTag secure:
    $InputFileStateFile stat-secure
    $InputFileSeverity Info
    $InputRunFileMonitor

    if $programname == 'secure' then @@100.104.102.184:514
    if $programname == 'secure' then stop

    ####################
    $InputFilePollInterval 180
    $InputFileName /var/log/yum.log
    $InputFileTag yum:
    $InputFileStateFile stat-yum
    $InputFileSeverity Info
    $InputRunFileMonitor

    if $programname == 'yum' then @@100.104.102.184:514
    if $programname == 'yum' then stop

########################################
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

6

TLS Overview and Configuration

Transport Layer Security (TLS) is used for end-to-end communication encryption.

TLS between a Recovery Appliance and client databases involves the use of certificates that authenticate and encrypt communication.

Certificates describe the server, who it belongs to, its connection string, etc. and is issued and signed by a trusted authority. Customers may choose third-party vendors or Oracle internal CA certificate authority.

For development and testing purpose, some customers choose to use self-signed certificate, which could be created by RACLI command.

- **Trusted Certificates** are generally obtained from a trusted Certified Authority (CA) through an application process (at the corporate level). These certificates are generally used between external systems. Because they were created by the CA, these certificates do not contain any local host names. The file type is `*.pem`.
- **Signed Certificates** are created as needed and contain the local host name as well as location and organization information as part of what authenticates it. These certificates are often used between local or internal systems. Signed certificates are specific to each Recovery Appliance. The file type is `*.p12`.

For TLS, both types of certificates are required.

This chapter provides general information on obtaining the certificates from a security website, as well as alternatively information on generating the certificates manually with RACLI commands. RACLI (`racli create certificate`) is a wrapper for **openssl** operations.

Whether obtained or generated, the created certificate is imported to the Recovery Appliance wallet using `racli add certificate` so that they are available for the network. Then, finally the

`racli alter network` establishes the needed encryption mode.

- **enable**: dual mode allows both encrypted and un-encrypted data.
- **only**: only encrypted data
- **disable**: only un-encrypted data

If replication is managed by RACLI, running `racli alter network` to disable or enable TLS on the downstream Recovery Appliance automatically imports the downstream Recovery Appliance's certificate into the upstream Recovery Appliance, updates the upstream Recovery Appliance's `tnsnames.ora`, and restarts the replication server. Thus, the manual steps mentioned in TLS Replication are no longer required. This is true for bidirectional replication as well.

If replication is not managed by RACLI, and the user runs `racli alter network`, then the manual steps described in TLS Replication are required.

Copyright © 2014, 2026, Oracle and/or its affiliates.

6.1 Certificate Management

This section describes in general terms the process after obtaining TLS certificates from a Certified Authority (CA) and management with the Recovery Appliance.

A **certificate authority (CA)** is a company or organization that acts to validate the identities of entities (such as websites, email addresses, companies, or individual persons) and bind them to cryptographic keys through the issuance of electronic documents known as **digital certificates**. A CA acts as a trusted third party for both the subject (owner) of the certificate and the party relying upon the certificate.

A digital certificate provides:

- **Authentication**; the certificate serves as a credential to validate the identity of its owner. In this case, it authenticates communication from the Recovery Appliance to its protected databases, to other replication Recovery Appliances, and to cloud archival storage.
- **Encryption** for secure communication over insecure networks such as the Internet.
- **Integrity** of documents **signed** with the certificate so that they cannot be altered by a third party in transit.

The format of these certificates is specified by the X.509 standard.

The techniques vary between CAs for validating the domain to prove that certificate applicant controls the given domain name.

Likewise each CA has its own application steps that are beyond the scope of this chapter to detail.

In general, upon completion of the certificate application process with your chosen CA, you (the applicant) downloads a bundle file containing all of your certificates.

The following assumes that you have that bundle file (`*.pfx`), named in this example `YourCompany.pfx`, generated by your chosen CA.

CA Bundle File (`*.pfx`) of Certificates

The Recovery Appliance TLS encryption requires both a trusted certificate (`*.pem`) and signed certificate (`*.p12`). Each certificate needs to be extracted from the bundle file (`*.pfx`) and then imported into the TLS wallet.

Certificate Creation Using Third Party Software

1. On the Recovery Appliance, obtain a list of all subject alternative names (SAN) by issuing the following command.

   ```
   racli list san
   ```

   > **Note**
   >
   > If this returns nothing, patch to a newer version.

2. With the SAN information available particularly for common name (CN) and DNS entries, go to your security website and enter this information to obtain the certificate package.

   Use the `PKCS#8` format and make sure to specify separate files.

Copyright © 2014, 2026, Oracle and/or its affiliates.

3. Download the certificate `ZIP` package.
4. Unzip the certificate `ZIP` package.

   The certificate `ZIP` package contains several files including trusted certificate and user certificate.

   - The trusted certificate may have `chain` or `root` in its name, and it is `*.pem` format.
   - User certificate is in `*.crt` format.
   - The `*.key` file should also in this directory from downloaded package.
5. With `openssl pkcs12`, sign the user certificate with the trusted certificate to create a `*.p12` file.

   ```
   openssl pkcs12 -export --in /<DIR>/<NAME>.crt
   --inkey /<DIR>/<NAME2>.key --certfile  /<DIR>/<NAME3>.pem
   --passin pass:<YOURPASSWORD> --passout pass: :<YOURPASSWORD>
   --out /<DIR>/<NAME4>.p12
   ```

   > **Note**
   >
   > Do not use `ewallet` or `cwallet` for `<NAME4>`. `<NAME4>` should refer to local host information or organization name used for `<NAME>`, `<NAME2>`, and `<NAME3>`.
6. Import both the trusted certificate and the signed user certificate into the Recovery Appliance wallet.

   ```
   racli add certificate --signed_cert=/<DIR>/<NAME4>.p12
   --trusted_cert=/<DIR>/<NAME3>.pem
   ```
7. Verify the certificates are in the Recovery Appliance wallet.

   ```
   racli list certificate
   ```
8. Continue with Configuring TLS Data Security on the Recovery Appliance followed by Configuring TLS Data Security on the Client.

6.1.1 Using Your Organization's CA process for TLS Certificates

This section details how to create TLS certificates using **openssl**.

Large organizations or government bodies, as examples, may have their own PKIs (public key infrastructure), each containing their own CAs.

For the case that your organization has its own certificate process, this section explains how to integrate Recovery Appliance certificates.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Prepare Information for the Certificates

1. On a Recovery Appliance as an `admin_user` or `root`, run this command.

   ```
   racli list san
   ```

   ```
   Created log /opt/oracle.RecoveryAppliance/log/racli_list_san.log
   Thu May  6 16:18:33 2021: Start: List SAN
   CN = zdlra09ingest-scan1.yourdomain.com
   DNS.1 = zdlra09adm01.yourdomain.com
   DNS.2 = zdlra09adm02.yourdomain.com
   DNS.3 = zdlra09ingest-scan1.yourdomain.com
   DNS.4 = zdlra09ingest01-vip.yourdomain.com
   DNS.5 = zdlra09ingest01.yourdomain.com
   DNS.6 = zdlra09ingest02-vip.yourdomain.com
   DNS.7 = zdlra09ingest02.yourdomain.com
   Thu May  6 16:18:39 2021: End: List SAN
   ```

   The `CN` (Common Name) item from your host is `<yourScanName>` which later corresponds to certificate files.

   In this example, `<yourScanName>` is "`zdlra09ingest-scan1`", the signed certificate file is `<yourScanName>.p12`, and the trusted certificate is `<yourScanName>.pem`.

2. Use an editor to create a `CRT` configuration file for **your organization's certification/ security process.**.

   In this example, it is named `<YOUR_CONFIG2>`. In your environment, all of the constructs with `YOUR_...` or `yourDir` are replaced with specific information from the local instance. And the `<YOUR_DNS>` items are replace with information obtained in previous step using `racli list san`.

   ```
   [req]
   default_bits = 2048
   prompt = no
   default_md = sha256
   req_extensions = v3_req
   distinguished_name = dn

   [ dn ]
   C=$args{YOUR_COUNTRY}
   ST=$args{YOUR_STATE}
   L=$args{YOUR_LOCATION}
   O=$args{YOUR_ORGANIZATION}
   OU=$args{YOUR_ORGANIZATION_UNIT}
   emailAddress=$args{YOUR_EMAIL_ADDRESS}
   CN = $list_san->{CN}
   [ v3_req ]
   keyUsage = keyEncipherment, dataEncipherment
   extendedKeyUsage = serverAuth
   subjectAltName = \@alt_names
   [alt_names]
   ```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
DNS.1 = <YOUR_DNS.1>
DNS.2 = <YOUR_DNS.2>
```

3. Upload the `<YOUR_CONFIG2>.CRT` file to **your organization's certification/security process.**

> **Note**
>
> When your CA organization generates the bundle:
>
> - Choose the format `PEM (OpenSSL)`.
> - Check the option for `including CRT file`, because `<YOUR_CONFIG2>` is needed.

4. From **your organization's certification/security process**, download the whole package to a location designated as `<yourDir>`. This example assumes it is named `<yourDownload>.crt`.

   The trusted certificate, `<yourDownload>.pem`, is within that package and is used from this package in later steps to generate a signed certificate.

5. On a Recovery Appliance as an `admin_user` or `root`, run this command to generate the key file. In this example, it is `yourScanName.key`.

```
openssl genrsa --passout pass:<yourPassword> --out <yourDir>/
<yourScanName>.key 2048
```

6. Get the certificate signed by the trusted certificate using the `pkcs12` format.

```
openssl pkcs12 --export --in <yourDir>/<yourDownload>.crt
    --inkey <yourDir>/<yourScanName>.key
    --certfile <yourDir>/<yourDownload>.pem
    --passin pass:<yourPassword>
    --passout pass:<yourPassword>
    --out <yourDir>/<yourScanName>.p12
```

7. Import the signed certificate into the TLS wallet.

```
racli add certificate --signed_cert=<yourDir>/<yourScanName>.p12
```

8. Import the trusted certificate into the TLS wallet.

```
racli add certificate --trust_cert=<yourDir>/<yourScanName>.pem
```

9. After importing the certificates, verify with "`racli list certificate`" that the certificates are in the `raa_certs` database table.

```
# racli list certificate

Created log /opt/oracle.RecoveryAppliance/log/
racli_list_certificate.20230329.1146.log
Wed Mar 29 11:46:49 2023: Start: List Certificate
Serial: 9A15CB4B76BBC52D
    Expire Time:      2024-03-28
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
    Certificate Type: trusted_cert

Serial: 95B9181340F644F0
    Expire Time:      2024-03-28
    Certificate Type: signed_cert

Wed Mar 29 11:46:49 2023: End: List Certificate
```

10. Continue with Configuring TLS Data Security on the Recovery Appliance followed by Configuring TLS Data Security on the Client.

6.1.2 Manually Creating TLS Certificates with RACLI

This section details how to create TLS certificates with RACLI.

If your organization does not have or is not using a Certificate Authority (CA), these instructions allow you to create the needed trusted and signed certificates for TLS operations.

The following information is required for both the trusted and signed certificates:

- Country Name
- State Name
- Organization Name
- Organization Unit Name
- Email Address

Generate Trusted and Signed Certificates using RACLI.

1. With the organization information on-hand, issue an RACLI command similar to:

```
# racli create certificate --country=US --state=CA --location=SF --
organization=oracle
--organization_unit=zdlra --email_address=<YOUR_EMAIL>

Created log /opt/oracle.RecoveryAppliance/log/
racli_create_certificate.20230329.1110.log
Enter New Password for Certificate:
Confirm New Password for Certificate:
Wed Mar 29 11:11:22 2023: Start: Create TLS Trusted Certificate
Wed Mar 29 11:11:26 2023: End: Create TLS Trusted Certificate
Wed Mar 29 11:11:26 2023: Start: Create TLS Signed Certificate
Wed Mar 29 11:11:31 2023: End: Create TLS Signed Certificate
Certificate(s) created under /raacfs/raadmin/config/cert
```

   The name of the certificate created is `<yourScanName>.p12`, where `<yourScanName>` is the CN for your environment.

2. To obtain `<yourScanName>` local host information and Common Name (CN) item.

```
racli list san

    Created log /opt/oracle.RecoveryAppliance/log/racli_list_san.log
    Thu May  6 16:18:33 2021: Start: List SAN
    CN = zdlra09ingest-scan1.yourdomain.com
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
DNS.1 = zdlra09adm01.yourdomain.com
DNS.2 = zdlra09adm02.yourdomain.com
DNS.3 = zdlra09ingest-scan1.yourdomain.com
DNS.4 = zdlra09ingest01-vip.yourdomain.com
DNS.5 = zdlra09ingest01.yourdomain.com
DNS.6 = zdlra09ingest02-vip.yourdomain.com
DNS.7 = zdlra09ingest02.yourdomain.com
Thu May  6 16:18:39 2021: End: List SAN
```

In this example, `<yourScanName>` is "`zdlra09ingest-scan1`" and the certificate file is `<yourScanName>.p12`.

You assign the certificate type (trusted or signed) later when added to the wallet.

Import Certificates into Wallet

Upon completion of creating the trusted and signed certificates, `<yourScanName>.pem` and `<yourScanName>.p12` respectively, import them into the Recovery Appliance wallet for TLS.

1. Import the certificates (trusted or signed) from the previous steps into the wallet. Here is the generic command, while specific examples are in the next steps.

   ```
   racli add certificate { [--trusted_cert=<VALUE>] |
         [--signed_cert=<VALUE>] | [--self_signed] }
   ```

   Arguments:

   - `--trusted_cert=<VALUE>`: Specify the full path and name of the trusted certificate to be added.
   - `--signed_cert=<VALUE>`: Specify the full path and name of the signed certificate in the trusted store to be added.
   - `--self_signed`: Specifies that Recovery Appliance will look for both certificates from designated locations. This should only be used when the certificates were created by "`racli create certificate`". This is not the Oracle recommended configuration, and is used only in a test environment.

   > **Note**
   >
   > Self-signed certificates should not be used long-term or for production. The recommendation is to use a (trusted) certificate signed by your Certification Authority.

2. Import the signed certificate into the TLS wallet. If the certificate was created by RACLI, include the `--self_signed` argument.

   ```
   racli add certificate --signed_cert=<yourDir>/<yourScanName>.p12 [--
   self_signed]
   ```

3. Import the trusted certificate into the TLS wallet. If the certificate was created by RACLI, include the `--self_signed` argument.

   ```
   racli add certificate --trust_cert=<yourDir>/<yourScanName>.pem [--
   self_signed]
   ```


Copyright © 2014, 2026, Oracle and/or its affiliates.

4. After importing the certificates, verify with "`racli list certificate`" that the certificates are in the `raa_certs` database table.

```
# racli list certificate

Created log /opt/oracle.RecoveryAppliance/log/
racli_list_certificate.20230329.1146.log
Wed Mar 29 11:46:49 2023: Start: List Certificate
Serial: 9A15CB4B76BBC52D
    Expire Time:      2024-03-28
    Certificate Type: trusted_cert

Serial: 95B9181340F644F0
    Expire Time:      2024-03-28
    Certificate Type: signed_cert

Wed Mar 29 11:46:49 2023: End: List Certificate
```

5. Continue with Configuring TLS Data Security on the Recovery Appliance followed by Configuring TLS Data Security on the Client.

After a certificate is in the `raa_certs` database table and when it has less than 90 validation days remaining, an incident is raised. If a certificate expires, the user is required to import a new valid certificate using RACLI to replace the old one.

- `racli add certificate`
- `racli remove certificate`

6.2 Configuring TLS Data Security on the Recovery Appliance

This section provides the steps for configuring TLS Data Security on the Recovery Appliance.

RACLI commands configure the TLS (Transport Layer Security). The Recovery Appliance these TLS modes:

- **only**: is `https` encryption alone.
- **enable**: is `http/https` dual mode.
- **disable**: is `http`, the default, without encryption.

The port numbers can be customized. The default ports for encryption are:

- TCPS: 2484
- HTTPS: 8005
- REPL_TCPS: 2485

The default ports for non-encrypted operation are:

- TCP: 1521
- HTTP: 8001
- REPL_TCP: 1522

Copyright © 2014, 2026, Oracle and/or its affiliates.

1. Verify with "`racli list certificate`" that the certificates are in the `raa_certs` database table.

```
# racli list certificate

Created log /opt/oracle.RecoveryAppliance/log/
racli_list_certificate.20230329.1146.log
Wed Mar 29 11:46:49 2023: Start: List Certificate
Serial: 9A15CB4B76BBC52D
    Expire Time:      2024-03-28
    Certificate Type: trusted_cert

Serial: 95B9181340F644F0
    Expire Time:      2024-03-28
    Certificate Type: signed_cert

Wed Mar 29 11:46:49 2023: End: List Certificate
```

2. To update the TLS mode on the Recovery Appliance that employs the certificates, issue a command similar to:

```
racli alter network --service=ra_server --encrypt=enable
```

> **Note**
>
> A complete Recovery Appliance outage is expected, because the whole CRS stack is restarted as part of the procedure. Additional steps are required because of this outage: pause replication, pause any backup scheduler, etc.

The general form of the command is:

```
racli alter network --service=ra_server
{ --encrypt=[enable|only|disable] }
[ --tcps_port=<VALUE>|--tcp_port=<VALUE> ]
[ --https_port=<VALUE>|--http_port=<VALUE> ]
[ --repl_tcp_port=<VALUE>|--repl_tcps_port=<VALUE>]
[ --silent ]
```

`--service`
Indicate the service being modified on the system. Valid value is "ra_server". Cannot be used of `--network_type` or its arguments.

`--network_type`
Indicate network type on the system. Cannot be used of `--service` or its arguments.

`--encrypt`
Specifies TLS encryption status on the system: "only" means HTTPS encryption; "enable" means dual HTTPS and HTTP; and "disable" means HTTP.

`--http_port`
Specifies HTTP port number to use. Default port is 8001.

Copyright © 2014, 2026, Oracle and/or its affiliates.

`--https_port`
Specifies HTTPS port number to use. Default port is 8005.

`--tcp_port`
Specifies the TCP port number to use. Default port is 1521.

`--tcps_port`
Specifies TCPS port number to use. Default port is 2484

`--rep_tcps_port`
Specifies the replication TCPS port number to use. Default port is 2485.

`--rep_tcp_port`
Specifies the replication TCP port number to use. Default port is 1522.

`--silent`
When present

3. Verify the health of the TLS.

```
# racli run check --check_name=tls_health
```

Changing TLS Encryption on the Recovery Appliance

The "`racli alter network`" command configures TCPS & HTTPS, and TCP & HTTP. It has three encryption modes of operation.

- **Enable TLS Encryption:** This enables dual mode TCP/TCPS and HTTP/HTTPS, and will use default ports unless otherwise specified.

```
racli alter network
--service=ra_server –-encrypt=enable
[ --tcps_port=<VALUE> ]
[ --https_port=<VALUE> ]
[ --repl_tcps_port=<VALUE> ]
```

- **Disable TLS Encryption:** This enables TCP and HTTP, and will use their default ports unless otherwise specified.

```
racli alter network
--service=ra_server –-encrypt=disable
[ --tcp_port=<VALUE> ]
[ --http_port=<VALUE> ]
[ --repl_tcp_port=<VALUE> ]
```

- **Enable Only TLS Encryption:** This enables only TCPS and HTTPS. The TCP and HTTP are disabled. Default ports are used unless otherwise specified.

```
racli alter network
--service=ra_server –-encrypt=only
[ --tcps_port=<VALUE> ]
[ --https_port=<VALUE> ]
[ --repl_tcps_port=<VALUE> ]
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

Validating TLS Usage

The following commands assist in monitoring the various TLS objects.

- `racli run check --check_name=tls_health`
- `racli run diagnostics --tag=tls`
- `racli run diagnostics --tag=tls_high`

6.3 Configuring TLS Data Security on the Client

This section provides the steps required to configure TLS Data Security on the Client (database).

The client requires some modifications to support TLS. The Recovery Appliance can use `https` encryption alone, in dual mode `http/https`, or without encryption `http`, the default.

Configuring Protected Databases to Support TLS

If you want to continue using non-TLS, update the RMAN settings by adding to `CONFIGURE CHANNEL DEVICE TYPE "_RA_NO_SSL=TRUE"`

```
CONFIGURE CHANNEL DEVICE TYPE
'SBT_TAPE' PARMS
'SBT_LIBRARY=<LIB_DIR>/libra.so,
ENV=(_RA_NO_SSL=TRUE,RA_WALLET=location=file:/<WRL>
     credential_alias==<DBNAME>_TCPS,_RA_TRACE_LEVEL=1000)' FORMAT '%U_%d';
```

An example `<LIB_DIR>` is `/u01/app/oracle/product/19.0.0.0/dbhome_1/lib`.

If you want to start using TLS, you need to perform the following steps.

1. Run verification to see where TLS presently is.

   ```
   racli run check --check_name=tls_health
   racli list certificate
   ```

2. Copy the trusted certificate (example: `raCA.pem`) from Recovery Appliance host to client side `<COPY_DIR>`.

   Permissions for the certificates should be `"oracle:oinstall"`.

3. Update wallet, or create new wallet. If existing wallet was created with `mkstore`, create a new wallet using `orapki` that can accept certificates. For example:

   ```
   orapki wallet create --wallet <WRL>
   ```

   ```
   orapki wallet create --wallet $ORACLE_HOME/dbs/Sydney
   ```


Copyright © 2014, 2026, Oracle and/or its affiliates.

4. Import the trusted certificate into the wallet from above.

```
orapki wallet add --wallet <WRL> --trusted_cert --cert <COPY_DIR>/
<NAME3>.pem
```

```
orapki wallet add --wallet $ORACLE_HOME/dbs/sydney --trusted_cert --
cert $ORACLE_HOME/dbs/sydney/raCA.pem
```

5. On a Recovery Appliance host, find the TCPS alias (example: `zdlra_tcps`) in `$ORACLE_HOME/network/admin/tnsnames.ora` file, copy it to `tnsnames.ora` file on client side.
6. Update wallet to `--auto_login`.

```
orapki wallet create --wallet <WRL> --auto_login
```

```
orapki wallet create --wallet $ORACLE_HOME/dbs/sydney --auto_login
```

7. Update the wallet with VPC user credentials for the new TCPS alias.

```
mkstore --wrl <WRL> --createCredential <DBNAME>_tcps <VPCUSER> <VPCPW>
```

```
mkstore --wrl $ORACLE_HOME/dbs/sydney --createCredential zdlra7_tcps
<VPCUSER> <VPCPW>
```

8. Add the wallet path `<WRL>` to `sqlnet.ora` file.
9. Validate on the client side using `tnsping`.

```
tnsping <DBNAME>_TCPS
```

```
tnsping ZDLRA7_TCPS
```

10. Connect to RMAN and update "`CONFIGURE CHANNEL DEVICE`" adding wallet information.

```
rman target / catalog <VPCUSER>/<VPCPW>@<DBNAME>_TCPS
```

```
rman target / catalog <VPCUSER>/<VPCPW>@zdlra7_tcps
```

Alternatively, after you type in the following command, you are prompted for the VPC user name and password.

```
rman target / catalog @<DBNAME>_TCPS
```

```
rman target / catalog @zdlra7_tcps
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

11. Validate the whole process by attempting to create a backup.

```
run
{     allocate CHANNEL c1 DEVICE TYPE 'SBT_TAPE' PARMS
"SBT_LIBRARY=<LIB_DIR>
        libra.so,ENV=(RA_WALLET='location=file:/<WRL>
        credential_alias=<DBNAME>_TCPS,RA_FORMAT=TRUE)";

        backup incremental level 1 filesperset 1 section size 64g database
plus archivelog not backed up filesperset 32;
        }
```

Validating TLS Usage

The following commands assist in monitoring the various TLS objects.

- `racli run check --check_name=tls_health`
- `racli run diagnostics --tag=tls`
- `racli run diagnostics --tag=tls_high`

6.4 Trouble-shooting TLS

This section provides some information about common TLS configuration errors.

If TLS isn't working, the following are items that can cause issues.

- The certificates are not correct.
  - Missing DNS information
  - Wrong format.
  - Certificate was not signed.
- The port is not open or available.
- The trusted certificate was not copied to the client side.
- The client side wallet is `mkstore` type that doesn't support certificate import.
- The RMAN settings were not updated after `tnsname` was updated and the certificate imported.
- The upstream Recovery Appliance wallets do not have certificates from the downstream Recovery Appliance.
- The upstream Recovery Appliance `tnsnames.ora` file does not have downstream Recovery Appliance new TCPS information.

Troubleshoot DNS

To obtain the DNS information, issues the following RACLI command on the Recovery Appliance.

```
racli list san
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

To check if the certificate has the the DNS information, make sure that the trusted certificate has no information and that the signed certificate has DNS information.

```
openssl x509 -text -noout -in cert.pem | grep  -i 'dns'
openssl x509 -text -noout -in <>.p12 | grep  -i 'dns'
```

Troubleshoot Certificates

Get certificate details from metadata table including type.

```
racli list certificate
```

Get certificate details from wallet.

```
orapki wallet display --wallet /raacfs/raadmin/config/ra_wallet/wallet/ --
complete
```

- User Certificates:
  - Subject: CN=<>-scan.subnet1.<>.oraclevcn.com

    > **Note**
    >
    > The scan address is not the same as the trusted certificate

  - Issuer: CN=Oracle DB Recovery Service Authority
- Trusted Certificates:
  - Subject: CN=Oracle DB Recovery Service Authority
  - Issuer: CN=Oracle DB Recovery Service Authority

Tips

- If the backup is not working and returns errors, check the certificates, wallet, and RMAN configuration settings.
- If the backup is hanging, check the Recovery Appliance's port. Make sure TCPS port (default 8005) is open on the Recovery Appliance.

  Check scan listener and the listener status of the Recovery Appliance.

6.5 TLS and Replication Setup

When replication between Recovery Appliances was established before configuring Transport Layer Security (TLS) for dual mode, the following additional steps are needed on **the upstream Recovery Appliance**.

> **Note**
>
> If TLS is configured before replication is established, no changes are required.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Copy the Trusted Certificate

1. Verify that TLS is not already set up on the Recovery Appliance.

   ```
   racli run check --check_name=tls_health
   racli list certificate
   ```

2. If replication is not being configured with an RACLI partnership that handles certificates, then these manual steps are required.

   Copy the trusted certificate to the upstream Recovery Appliance. Permissions for the certificates should be "`oracle:oinstall`".

   > **Note**
   >
   > Do not replace existing certifications on the upstream Recovery Appliance, which may have the same names. Use a temporary location, such as `/tmp` to prevent the overwriting of the existing default certificates at `/raacfs/raadmin/config/cert/raCA.pem`.

   If the upstream Recovery Appliance is TLS enabled and shares the same trusted certification as the downstream Recovery Appliance, skip the following steps and go to the steps for updating `tnsnames.ora`.

3. If the upstream and downstream Recovery Appliance replication was established *without using RACLI partnerships.*, then these manual steps to create the correct replication wallet are required.

   > **Note**
   >
   > If the upstream and downstream Recovery Appliance replication was established using RACLI, skip this step, because the correct replication wallet should already exist.

   To check that replication is managed through RACLI, run:

   ```
   racli status replication_server [--all]
   racli list replication_server [--all]
   ```

   If creating the wallet is required, the new wallet location needs to be `/raaacfs/raadmin/replication/orapki/` to be compliant with RACLI replication management. The following command asks for the wallet password and the pkcs12 certificate password.

   ```
   orapki wallet create -wallet /raaacfs/raadmin/replication/orapki/ -
   auto_login
   ```

4. Import the trusted certificate into the wallet. Here is the generic command.

   ```
   orapki wallet add -wallet /raaacfs/raadmin/replication/orapki/  --
   trusted_cert
   --cert <CERT_DIR>/<NAME>.pem
   ```


Copyright © 2014, 2026, Oracle and/or its affiliates.

For one-way replication, the command is only issued on the upstream Recovery Appliance. For bi-directional replication, the command is issued on both Recovery Appliances.

The command asks for the wallet password and the pkcs12 certificate password. Existing credentials can be checked from the previous replication wallet with:

```
$ mkstore -wrl /raacfs/raadmin/replication/orapki/ -listCredential
$ mkstore -wrl /raacfs/raadmin/replication/orapki/ -list
$ mkstore -wrl /raacfs/raadmin/replication/orapki/ -viewEntry
oracle.security.client.password1
```

Update `tnsnames.ora` and the Replication Server

1. Update `tnsnames.ora` file with the TCPS info.

   This can be copied from `tnsnames.ora` file on the Recovery Appliance.

   TCPS service for each Recovery Appliance is added in RDBMS `ORACLE_HOME/network/Admin/tnsnames.ora` when "`racli alter network --service=ra_server --encrypt=enable`" is run.

2. If a new wallet was created manually above, pause the replication server, update the replication server accordingly, and then resume the replication server.

   - If using RACLI (23.1 and later):

     ```
     racli stop replication_server
     ```

     ```
     racli update replication_server
     ```

     ```
     racli start replication_server
     ```

   - If using the DBMS_RA (21.1 and earlier):

     PAUSE_REPLICATION_SERVER

     UPDATE_REPLICATION_SERVER

     RESUME_REPLICATION_SERVER

Copyright © 2014, 2026, Oracle and/or its affiliates.

7

Managing Protection Policies with Recovery Appliance

This chapter explains how to manage protection policies and polling policies, which are part of "Setup and Configuration for Recovery Appliance".

This chapter contains the following topics:

- About Protection Policies
- Creating a Backup Polling Policy (Command-Line Only)
- Creating a Protection Policy
- Updating a Protection Policy
- Deleting a Protection Policy

7.1 About Protection Policies

A protection policy is the central mechanism for controlling management of backup storage space, based on pre-defined recovery window goals. From the perspective of a DBA, the most important elements of a protection policy are the disk and tape recovery windows.

This section contains the following topics:

- Purpose of Protection Policies
- Overview of Protection Policies
- User Interfaces for Protection Policies
- Basic Tasks for Managing Protection Policies

> **See Also**
>
> "Protection Policies" for an architectural overview

7.1.1 Purpose of Protection Policies

For every database associated with it, a protection policy specifies:

- The recovery window goal for disk backups
- The recovery window for tape backups
- Whether Recovery Appliance must replicate backups or copy them to tape before deleting them
- Which Recovery Appliance storage location is used for backups
- An optional backup polling policy

Copyright © 2014, 2026, Oracle and/or its affiliates.

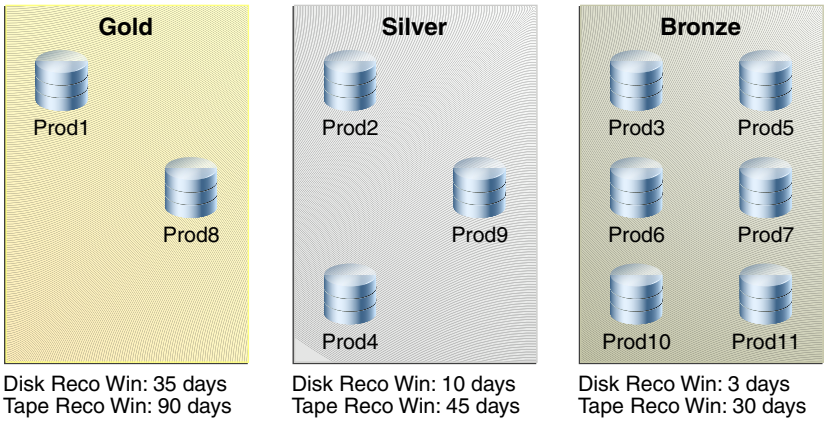
You can attach multiple protected databases to a single protection policy. A Recovery Appliance may have a variety of protection policies to support different data protection support levels. For example, protection policies can be generic service levels such as gold, silver, and bronze. Alternatively, policies can be specific to the requirements of protected databases and applications.

7.1.2 Overview of Protection Policies

A protection policy is a named, logical object recorded in the Recovery Appliance metadata database. To be added to a Recovery Appliance, a protected database must be associated with a specific protection policy. The default protection polices are Platinum, Gold, Silver, and Bronze.

Each protection policy specifies different values for the disk and tape recovery windows. These values apply to every database protected by the policy. For example, Figure 7-1 shows three of the default protection policies, with different protected databases assigned to each policy. In the example, databases `prod3` and `prod11` are in the same policy, and so both have the same disk recovery window goal of 3 days.

Figure 7-1 Protection Policies



7.1.3 Guidelines for Protection Policies

Here are several considerations to create effective protection policies.

- All databases in a protection policy must share the following:
  - Recovery Window Compliance (14 days / 30 days / etc.). This should be smaller than Recovery Window Goal. The Recovery Window Compliance may be null. If too large, this can result in the Recovery Appliance rejecting new backups, because old backups for compliance purposes have not *"expired"* yet and made their storage space available for re-use with incoming backups.
  - Recovery Window Goal (14 days / 30 days / etc.). This is a goal to strive for and helps determine amount of storage required. However, if the amount of free storage becomes too small, the oldest backups might have their storage space reclaimed for new backups. In such a case, the goal isn't met but continued operation and the receiving of incoming backups is not prevented. This is the difference from the recovery window compliance.

Copyright © 2014, 2026, Oracle and/or its affiliates.

  - Max Disk Retention (default / 21 days / 35 days / etc.)
  - Tape Retention Policy (90 Days / 365 Days / 7 years)
  - Tape Operation Schedule (Sunday Full / Daily Incremental / Daily ARCH)
  - Replication Configuration (Replicate or No-Replicate, and which Recovery Appliances to replicate to)
- If a production database needs to be replicated but a development database does not, this case requires two (2) protection policies.

  Similarly, if a production database needs to be replicated but another production database does not, this case also requires two (2) protection policies.
- Geographical regions or different lines of business can mean additional protection policies. For example, the regions of North America and Europe might require two (2) protection policies.
- Tape operations that occur on different days requires a protection policy for each day.

  For example, if due to volume, certain databases perform their weekly full backup on Sunday and others on Monday, this requires two (2) protection policies. If all databases perform their weekly full backup on Sunday, then only one (1) protection policy is needed.
- If the number of days for tape retention is different between two databases, this requires two (2) protection policies.

A protection policy is a named, logical object recorded in the Recovery Appliance metadata database. To be added to a Recovery Appliance, a protected database must be associated with a specific protection policy. The default protection polices are Platinum, Gold, Silver, and Bronze.

Each protection policy specifies different values for the disk and tape recovery windows. These values apply to every database protected by the policy. For example, Figure 7-1 shows three of the default protection policies, with different protected databases assigned to each policy. In the example, databases `prod3` and `prod11` are in the same policy, and so both have the same disk recovery window goal of 3 days.

As an example of an update to a protection policy, the customer may choose to change the `LOG_COMPRESSION_ALGORITHM` setting in a protection policy for generally one or both of the below reasons:

- Reduction of CPU utilization on the appliance attributed to creation and compression of archived log backups.
- Reduction of CPU utilization on the protected database during recovery operations, attributed to decompression of archived log backups before the logs can be applied on the restored data files.

Although Oracle cannot provide detailed CPU utilization and compression ratio differences between the different algorithms, as they are highly data type dependent, generally:

- `LOW` and `MEDIUM` settings utilize less CPU than `BASIC` and `HIGH` for performing compression/ decompression, with trade-off of lower compression ratio (i.e. higher space usage on appliance).
- `MEDIUM` offers the optimal balance of CPU consumption and compression ratio in most cases.
- `LOW` offers the least CPU consumption, at the expense of a modest reduction in compression (higher space usage on appliance) ratio compared to `MEDIUM` and `BASIC`.
- `OFF` disables the compression.

Copyright © 2014, 2026, Oracle and/or its affiliates.

If a significant increase of space is noticed then the `LOG_COMPRESSION_ALGORITHM` can be changed back to `BASIC`.

The `HIGH` setting is not recommended due to significant CPU consumption.

For more details on log compression usage, see *"Changes in the Protection Policy Compression Algorithms"* (MOSnote Doc ID KB392785).

When a protection policy has `SECURE_MODE` set to `YES`, then backups that are not encrypted are rejected before they can be uploaded to the Recovery Appliance, by design. When redo logs are being shipped directly to the Recovery Appliance, they also must be encrypted. However, the check for redo encryption happens *after* the redo log completes, so future attempts to open a new log on the Recovery Appliance are rejected. A few logs might get started before the archived log destination status shows redo being rejected. This condition clears when an encrypted redo log backup is sent to the Recovery Appliance. After which, future redo log switch are accepted on the Recovery Appliance.

7.1.4 User Interfaces for Protection Policies

This section contains the following topics:

- Accessing the Create Protection Policy Page in Cloud Control
- DBMS_RA Procedures Relating to Protection Policies
- Recovery Catalog Views for Protection Policies

7.1.4.1 Accessing the Create Protection Policy Page in Cloud Control

The Create Protection Policy page in Oracle Enterprise Manager Cloud Control (Cloud Control) is the recommended interface for creating protection policies.

To access the Create Protection Policy page:

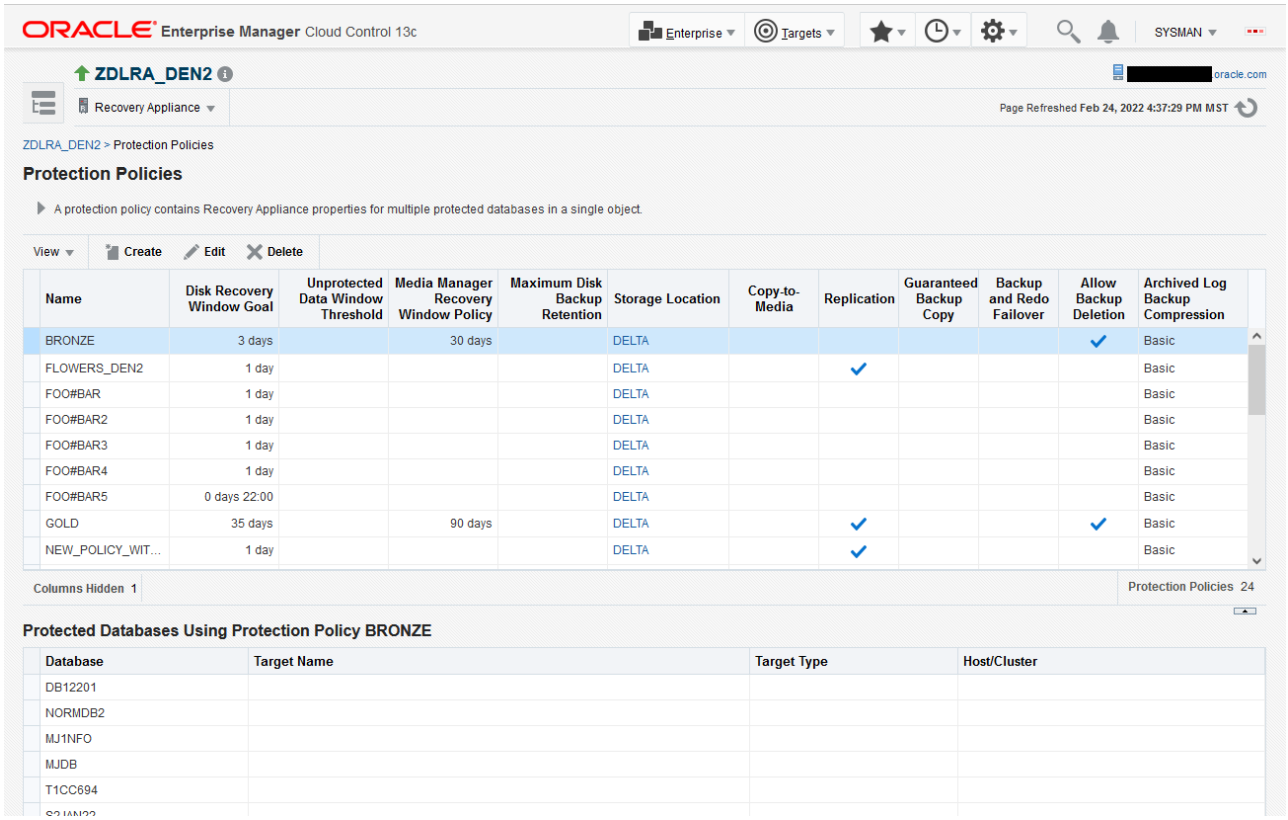
1. Access the Recovery Appliance Home page, as described in "Accessing the Recovery Appliance Home Page".
2. From the **Recovery Appliance** menu, select **Protection Policies**.

   The Recovery Appliance Login page appears.
3. Enter your login credentials, and then click **Login**.

   The Protection Policies page appears, as shown in the example in Figure 7-2.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Figure 7-2 Protection Policies Page



7.1.4.2 DBMS_RA Procedures Relating to Protection Policies

You can use the `DBMS_RA` package to create and manage protection policies. Table 7-1 describes the principal program units relating to protection policies.

Table 7-1 DBMS_RA Protection Policy Procedures

Program Unit	Description
CREATE_POLLING_POLICY	Creates a backup polling policy.
CREATE_PROTECTION_POLICY	Creates a protection policy.
DELETE_PROTECTION_POLICY	Deletes a protection policy.
UPDATE_PROTECTION_POLICY	Updates a protection policy.

> **See Also**
>
> DBMS_RA Package Reference

7.1.4.3 Recovery Catalog Views for Protection Policies

You can monitor protection policies using the Recovery Appliance catalog views. Table 7-2 summarizes the views that are most relevant for protection policies.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 7-2 Recovery Catalog Views for Protection Policies

View	Description
RA_PROTECTION_POLICY	This view describes the defined protection policies.
RA_POLLING_POLICY	This view describes the defined backup polling policies.
RA_DATABASE	The POLICY_NAME column of this view lists the protection policy used by this protected database.
RA_REPLICATION_CONFIG	The PROTECTION_POLICY column of this view lists the protection policy for a particular Recovery Appliance used for replication.
RA_REPLICATION_DATABASE	The POLICY_NAME column of this view lists the protection policy that is associated with this replication server and database.
RA_REPLICATION_PAIR	This view lists replication information for replicating protection policies.
RA_REPLICATION_POLICY	The POLICY_NAME column of this view lists the protection policy associated with this replication server configuration.

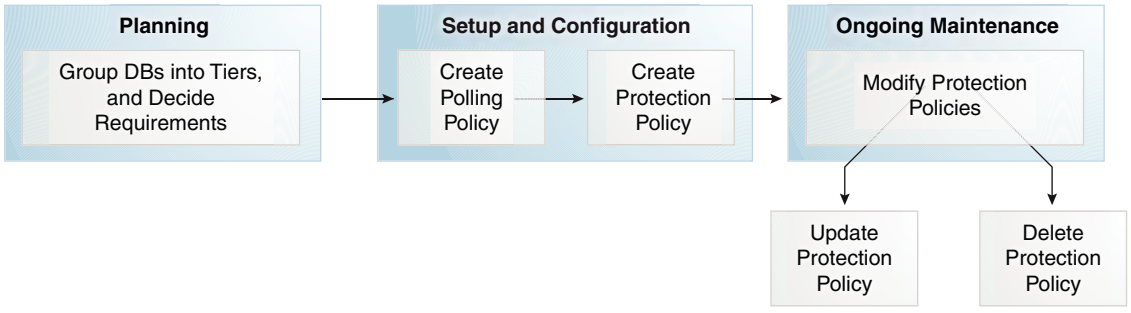
See Also

Recovery Appliance View Reference

7.1.5 Basic Tasks for Managing Protection Policies

This section explains the basic tasks involved in managing protection policies. Figure 7-3 shows the overall workflow described in Recovery Appliance Workflow , with the protection policy tasks highlighted.

Figure 7-3 Protection Policy Tasks in Recovery Appliance Workflow



Typically, you perform protection policy tasks in the following sequence:

1. During the planning phase, group the databases into tiers, and decide the recovery requirements for each tier.

   "Planning for Recovery Appliance" describes these tasks.

2. During the configuration phase (see "Setup and Configuration for Recovery Appliance"), create one protection policy for each database tier.

Copyright © 2014, 2026, Oracle and/or its affiliates.

a. Optionally, if your Recovery Appliance has access to a backup polling location, and if you are performing configuration using command-line tools, then create a backup polling policy.

"Creating a Backup Polling Policy (Command-Line Only)" describes this task.

> **Note**
>
> Cloud Control enables you to configure the polling policy and the protection policy in the same page.

b. Create a protection policy for a specific database tier.

"Creating a Protection Policy" describes this task.

3. During the ongoing maintenance phase (see "Maintenance Tasks for Recovery Appliance"), modify protection policies as needed. Typical modification tasks include:
   - Update the attributes of a protection policy.

     "Updating a Protection Policy" describes this task.
   - Delete a protection policy.

     "Deleting a Protection Policy" describes this task.

7.2 Protection Policy Attributes

A protection policy is created with the `DBMS_RA.CREATE_PROTECTION_POLICY` procedure or with Cloud Control. The protection policy sets some of the following attributes for all protected databases assigned to it: Some attributes are mutually exclusive. The following is a representative list of attributes to consider in new protection policies.

Table 7-3 Protection Policy Attributes (subset)

Attribute	Description
`storage_location_name`	A Recovery Appliance storage location for storing backups.
`polling_policy_name`	An optional backup polling policy that determines whether Recovery Appliance polls a storage location for backups
`recovery_window_goal`	The disk recovery window goal for the protected database.
`recovery_window_sbt`	The SBT retention period for the protected database.
`guaranteed_copy`	The guaranteed copy setting, which determines whether backups protected by this policy must be copied to tape or cloud before being considered for deletion.
`allow_backup_deletion`	Setting this to `NO` will prevent RMAN users from deleting backups on the Recovery Appliance, necessary for compliance rules. The default value is set to YES.
`store_and_forward`	The setting for the Backup and Redo Failover feature. This setting is used only in a protection policy defined on the alternate Recovery Appliance where the protected databases associated with this policy will redirect backups and redo in the event of an outage on the primary Recovery Appliance.
`max_retention_window`	The maximum length of time that the Recovery Appliance retains backups for databases that use this retention policy.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 7-3 (Cont.) Protection Policy Attributes (subset)

Attribute	Description
unprotected_window	The maximum acceptable difference between the current time and the latest time that the database can be restored.
autotune_reserved_space	This setting is used to control whether the Recovery Appliance will automatically define and update the reserved_space settings for databases associated with this policy.
recovery_window_compliance	This setting specifies a time range for each database backup in which backups will not be deleted. This value must be equal to or smaller than recovery_window_goal. Too large a value can result in filling disk_reserved_space with compliance protected backups, whereby new backups are then rejected.
keep_compliance	This setting prevents an administrator from using RMAN CHANGE command to shrink the "keep until time" specified for an archival backup. If KEEP_COMPLIANCE is YES, KEEP FOREVER backups will never be deleted. NO means the "keep until time" for an archival backup may be modified by the RMAN CHANGE command. NO is the default.
max_reserved_space	The maximum disk_reserved_space setting permitted for each database in the protection policy. The format of this value is a character string that must contain a number consisting only of the characters 0-9, followed optionally by one of the following unit specifiers: If max_reserved_space is specified as NULL, the max_reserved_space setting for databases defaults to 2 x disk_reserved_space.
secure_mode	Determines whether backups stored on the Recovery Appliance must be encrypted. YES means that only encrypted backup and redo are accepted by the Recovery Appliance. NO means unencrypted backups are allowed to be stored on the Recovery Appliance. NO is the default.
level0_refresh	If specified, the Recovery Appliance chooses some number of data files from each backup to be level 0 backups. This spreads the creation of new level 0 backup data across the level0_refresh interval. Specify the refresh cycle as any valid INTERVAL DAY TO SECOND expression, such as INTERVAL '20' DAY (20 days). If you set the value to 100 days, then 1% of the database will perform a level 0 backup each day. Effectively, once 100 days is complete, all datafiles will be level 0 with RA_FORMAT=TRUE (or *Block Pool Formatted*). The purpose of this option is to limit the number of Data Encryption Key (DEK) hashes required to restore a database. Each level 1 backup has a new DEK. During a restore, every block is sent to the client along with all the DEKs that are associated with them. During ingest, the DEKs are counted. When 65% of the maximum DEKs for the buffer is reached, a new level 0 refresh is set for that datafile.

You can associate an optional replication server configuration with a protection policy. The replication configuration applies to all protected databases associated with the protection policy.

Copyright © 2014, 2026, Oracle and/or its affiliates.

When a protection policy has SECURE_MODE set to YES, then backups that are not encrypted are rejected before they can be uploaded to the Recovery Appliance, by design. When redo logs are being shipped directly to the Recovery Appliance, they also must be encrypted. However, the check for redo encryption happens *after* the redo log completes, so future attempts to open a new log on the Recovery Appliance are rejected. A few logs might get started before the archived log destination status shows redo being rejected. This condition clears when an encrypted redo log backup is sent to the Recovery Appliance. After which, future redo log switch are accepted on the Recovery Appliance.

Note

Before release 21.1, any backup copy anywhere (tape or cloud) counted as a copy for a backup and would allow for deletion on the Recovery Appliance. If you had both cloud and tape, you might have incomplete backups on either cloud and tape, but the Recovery Appliance would incorrectly consider the set copied. Further with replication, the backups could be deleted on the downstream Recovery Appliance, leave backups never copied, and thus never released by the upstream Recovery Appliance.

After release 21.1, the guaranteed_copy attribute was added to the library. When guaranteed_copy is set on the library, the Recovery Appliance will not directly delete the copy in the library. [The tape/cloud manager shouldn't delete the copy either.] Each library with the guaranteed_copy attribute must have a copy of a given backup before it is eligible for deletion from the Recovery Appliance.

The APIs create_protection_policy and update_protection_policy check whether a guaranteed_copy library/template/attribute_set was available to the protection_policy before the protection_policy could have guaranteed_copy set. Other improvements protect the changing of libraries, templates, or attribute_set against the last removal of a library/template/attribute_set path from a protection_policy with the guaranteed_copy attribute set.

7.3 Creating a Protection Policy

This section explains how to create a protection policy using either Cloud Control (recommended) or the DBMS_RA.CREATE_PROTECTION_POLICY procedure. The best practice is to create a separate protection policy for each tier of databases (see "Task 1: Group protected databases into tiers").

You must be logged in to the Recovery Appliance as RASYS or as a named db_user with user_type=admin.

- Cloud_Control
- Command_Line

Cloud_Control

To create a Protection Policy with Cloud Control:

1. From any Cloud Control page, select **Targets**, and then **Recovery Appliances**.

   The Recovery Appliances page appears.

Copyright © 2014, 2026, Oracle and/or its affiliates.

2. From the **Recovery Appliances** drop-down menu, select **Protection Policies**.

   The **Protection Policies** page has two areas. The upper table lists the protection policies available, and has controls for **Create**, **Edit**, and **Delete**.

   When a policy is highlighted in the upper table, the lower table lists the protected databases that use that particular policy.

3. Click **Create**.

   The **Create Protection Policy** page appears.

Figure 7-4 Create Protection Policy Page



In this page, the default Recovery Appliance storage location `DELTA` is already selected.

Copyright © 2014, 2026, Oracle and/or its affiliates.

4. Enter values as follows:
   - In the **Name** field, enter the name of the new protection policy.

     For example, enter `bronze_dev`.
   - In the **Description** field, enter a description for the new policy.

     For example enter, `Policy with disk recovery window of 3 days and no tape backup`.
   - In the **Recovery Window** field of the **Disk Recovery Window Goal** section, specify a recovery window goal that the Recovery Appliance should attempt to meet for point-in-time recovery using disk backups, and then select the units.

     For example, enter 3 and then select **days**.
   - If the protection policy is being configured for regulatory operation, specify the **Recovery Window Compliance**. This setting specifies a time range for each database backup in which backups will not be deleted. This value must be equal to or smaller than `recovery_window_goal`. Too large a value can result in filling `disk_reserved_space` with compliance protected backups, whereby new backups are then rejected.
   - In the **Threshold** field of the **Unprotected Data Window Threshold** section, enter the maximum tolerable interval for data loss.

     For example, enter 5 and then select **minutes**.
   - If the protection policy is being configured for compliance operation, **Keep Compliance** specifies that backups are held until their "keep until time".
   - The **Autotuned Reserved Space** specifies whether or not the Recovery Appliance will automatically define and / or update the `reserved_space` for databases associated with this policy. When `autotune_reserved_space` is enabled, backups for compliance databases, which previously would have been rejected due to a lack of reserved space, automatically increase `reserved_space` to be able to accept the backups.
   - In the **Recovery Window** field of the **Media Manager Recovery Window Policy** section, optionally specify a larger window within which point-in-time recovery from a media manager will be maintained.

     For example, if no tape backup is desired, then leave this field blank.
   - In the **Maximum Retention** field of the **Maximum Disk Backup Retention** section, enter the maximum time that the Recovery Appliance must retain disk backups. This should be greater than or equal to the recovery window goal.

     For example, if not specified, backups are retained beyond the disk recovery goal as space permits.
5. Optionally, you can change items in the **Advanced** section.
   - In the **Backup and Redo Failover** section, specify whether protection databases associated with this protection policy are using this Recovery Appliance as an alternate destination.
   - If not using Recovery Window Compliance or Keep Compliance, then optionally the **Backup Deletion** section specifies if the Recovery Appliance allows backup deletion with `RMAN DELETE` (administrator role).
   - In the **Backup Polling Location** section, define a backup polling policy.
     - In the **Location** field, specify a directory accessible by the Recovery Appliance.
     - In the **Frequency** field, specify a time interval, and then select the time units.

Copyright © 2014, 2026, Oracle and/or its affiliates.

For example, to specify that backup polling is disabled, leave the fields blank.

- In the **Backup Copy Policy** section, specify whether the Recovery Appliance must replicate backups or copy backups to tape before deleting them.

  For example, select **Always accept new backups even if it requires purging existing backups not yet copied to tape or replicated**.

- In the **Archived Log Backup Compression**, you can select the compression algorithm for archived log backups.

6. Click **OK**.

   A deployment procedure is submitted to create the protection policy. A confirmation message to this effect is displayed at the top of the protection policies page. After the Protection Policies pages is refreshed, the newly created policy appears in the list.

See Also

- "How Recovery Appliance Manages Storage Space"
- "Backup Polling Policies"
- "Backup Polling Locations"
- Cloud Control online help for more information about the Create Protection Policy page

Command_Line

To create a Protection Policy with PL/SQL:

Assume that you want to create a protection policy named `bronze_dev` for a tier of databases in a development environment. You have the following requirements for all databases protected by this policy:

- The disk recovery window goal is 3 days, which means that every database must be recoverable using disk backups to any time within the last 3 days, starting from the current time.
- You do not want to archive backups to tape.
- You want the Recovery Appliance to receive new backups even if old backups must be deleted because available storage space is low.
- No backup polling policy is enabled.

You also want to create policies for `gold_dev`, with a disk recovery window goal of 35 days, and `silver_dev`, with a disk recovery window goal of 10 days. Additionally, you create one policy named `bronze_dev` with a disk recovery window goal of 12 hours.

1. Start SQL*Plus or SQL Developer, and then log in to the metadata database as `RASYS` or as a named `db_user` with `user_type=admin`.
2. Run the `DBMS_RA.CREATE_PROTECTION_POLICY` procedure.

   For example, execute the following PL/SQL anonymous block:

```
BEGIN
 DBMS_RA.CREATE_PROTECTION_POLICY (
  protection_policy_name => 'bronze_dev',
  description            => 'For protected dbs in bronze tier',
  storage_location_name  => 'delta',
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
  recovery_window_goal   => INTERVAL '3' DAY,
  guaranteed_copy        => 'NO');
 DBMS_RA.CREATE_PROTECTION_POLICY (
  protection_policy_name => 'silver_dev',
  description            => 'For protected dbs in silver tier',
  storage_location_name  => 'delta',
  recovery_window_goal   => INTERVAL '10' DAY,
  guaranteed_copy        => 'NO');
 DBMS_RA.CREATE_PROTECTION_POLICY (
  protection_policy_name => 'gold_dev',
  description            => 'For protected dbs in gold tier',
  storage_location_name  => 'delta',
  recovery_window_goal   => INTERVAL '35' DAY,
  guaranteed_copy        => 'NO');
 DBMS_RA.CREATE_PROTECTION_POLICY (
  protection_policy_name => 'test_dev',
  description            => 'Test policy',
  storage_location_name  => 'delta',
  recovery_window_goal   => INTERVAL '12' HOUR,
  guaranteed_copy        => 'NO');
END;
```

Note

Pay attention to the attributes that are mutually exclusive, such as the parameters associated with compliance versus the parameters of back up deletion or autotuning reserved space.

3. Optionally, query the recovery catalog to confirm creation of the policy.

   For example, query `RA_PROTECTION_POLICY` as follows (sample output included):

```
COL POLICY_NAME FORMAT a11
COL DESCRIPTION FORMAT a36
SELECT POLICY_NAME, DESCRIPTION,
       TO_CHAR(EXTRACT(DAY FROM RECOVERY_WINDOW_GOAL),'fm00')||':'||
       TO_CHAR(EXTRACT(HOUR FROM RECOVERY_WINDOW_GOAL),'fm00')||':'||
       TO_CHAR(EXTRACT(MINUTE FROM RECOVERY_WINDOW_GOAL),'fm00')||':'||
       TO_CHAR(EXTRACT(SECOND FROM RECOVERY_WINDOW_GOAL),'fm00')
         AS "DD:HH:MM:SS"
FROM   RA_PROTECTION_POLICY
WHERE  POLICY_NAME LIKE '%DEV'
ORDER BY POLICY_NAME;

POLICY_NAME DESCRIPTION                          DD:HH:MM:SS
----------- ------------------------------------ ---------------
BRONZE_DEV  For protected dbs in bronze_dev tier 03:00:00:00
GOLD_DEV    For protected dbs in gold_dev tier   35:00:00:00
SILVER_DEV  For protected dbs in silver_dev tier 10:00:00:00
TEST_DEV    Test policy                          00:12:00:00
```

See Also

- "Guaranteed Copy"
- "Backup Polling Policies"

Copyright © 2014, 2026, Oracle and/or its affiliates.

7.4 Updating a Protection Policy

This section explains how to update protection policies using either Cloud Control (recommended) or the DBMS_RA PL/SQL package.

You must be logged in to the metadata database as RASYS or as a named db_user with user_type=admin. The protection policy must exist.

In this example, the protection policy is bronze_dev and is changing its disk recovery window goal from 3 days to 6 days.

- Cloud_Control
- Command_Line

Cloud_Control

To update a protected database with Cloud Control:

1. Access the Protection Policies page, as described in "Accessing the Create Protection Policy Page in Cloud Control".
2. In the Protection Policies table, select the protection policy that you want to edit.

   For example, select the BRONZE_DEV row.
3. Click **Edit**.

   The Edit Protection Policy page appears.
4. Change the desired values, and then click **OK**.

   For example, in the **Recovery Window** field of the Disk Recovery Window Goal section, enter 6.

   The Protection Policies page appears, with the newly updated protection policy listed.

Command_Line

To update a protection Policy with DBMS_RA:

To update a protection policy, execute the DBMS_RA.UPDATE_PROTECTION_POLICY procedure. Parameters that are null retain their existing values. For example, if guaranteed_copy is currently NO for a protection policy, and if you specify null for this parameter in DBMS_RA.UPDATE_PROTECTION_POLICY, then the value remains NO.

1. Start SQL*Plus or SQL Developer, and then log in to the metadata database as RASYS or as a named db_user with user_type=admin.
2. Run the DBMS_RA.UPDATE_PROTECTION_POLICY procedure.

   For example, execute the following PL/SQL anonymous block:

```
BEGIN
 DBMS_RA.UPDATE_PROTECTION_POLICY(
  protection_policy_name => 'bronze_dev',
  recovery_window_goal   => INTERVAL '6' DAY);
END;
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

3. Optionally, query the recovery catalog to confirm the update of the policy.

   For example, query `RA_PROTECTION_POLICY` as follows (sample output included):

```
COL POLICY_NAME FORMAT a11
COL DESCRIPTION FORMAT a36
SELECT POLICY_NAME, DESCRIPTION,
       TO_CHAR(EXTRACT(DAY FROM RECOVERY_WINDOW_GOAL),'fm00')    ||':'||
       TO_CHAR(EXTRACT(HOUR FROM RECOVERY_WINDOW_GOAL),'fm00')   ||':'||
       TO_CHAR(EXTRACT(MINUTE FROM RECOVERY_WINDOW_GOAL),'fm00') ||':'||
       TO_CHAR(EXTRACT(SECOND FROM RECOVERY_WINDOW_GOAL),'fm00')
         AS "DD:HH:MM:SS"
FROM   RA_PROTECTION_POLICY
WHERE  POLICY_NAME='BRONZE_DEV';

POLICY_NAME DESCRIPTION                          DD:HH:MM:SS
----------- ------------------------------------ ---------------
BRONZE_DEV  For protected dbs in bronze tier     06:00:00:00
```

7.5 Deleting a Protection Policy

This section explains how to delete protection policies using either Cloud Control (recommended) or the `DBMS_RA` PL/SQL package.

You must be logged in to metadata database of the Recovery Appliance as `RASYS` or as a named `db_user` with `user_type=admin`.

The protection policy must not be associated with any protected database. To delete a policy that is associated with one or more databases, you must associate those databases with different policies before you can delete the desired policy.

In the following example, assume that you want to delete the `bronze_dev` policy.

- Cloud_Control
- Command_Line

Cloud_Control

To delete a Protection Policy with Cloud Control:

1. Access the Protection Policies page, as described in "Accessing the Create Protection Policy Page in Cloud Control".
2. In the Protection Policies table, select the protection policy that you want to delete.

   For example, select the `BRONZE_DEV` row.
3. Click **Delete**.

   A confirmation window appears.
4. Click **Yes**.

   The Protection Policies page appears, with the deleted protection policy no longer listed.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Command_Line

To delete a Protection Policy with DBMS_RA:

In the following example, assume that you want to delete the `bronze_dev` policy.

1. Start SQL*Plus or SQL Developer, and then log in to the metadata database as `RASYS` or as a named `db_user` with `user_type=admin`..
2. Confirm that the protection policy that you intend to delete is not currently associated with any protected databases.

   For example, query all protection policies not associated with databases:

   ```
   SELECT POLICY_NAME AS "Currently unused policy"
   FROM   RA_PROTECTION_POLICY
   WHERE  POLICY_NAME NOT IN (SELECT POLICY_NAME FROM RA_DATABASE)
   ORDER BY POLICY_NAME;

   Currently unused policy
   -----------------------
   BRONZE_DEV
   ```

3. Delete the policy.

   For example, execute the following PL/SQL anonymous block to delete the protection policy named `bronze_dev`:

   ```
   BEGIN
     DBMS_RA.DELETE_PROTECTION_POLICY(
       protection_policy_name => 'BRONZE_DEV');
   END;
   ```

4. Optionally, confirm the deletion.

   For example, count the rows for policies named `bronze_dev` (sample output included):

   ```
   SELECT COUNT(*)
   FROM   RA_PROTECTION_POLICY
   WHERE  POLICY_NAME = 'BRONZE_DEV';

     COUNT(*)
   ----------
            0
   ```

7.6 Creating a Backup Polling Policy (Command-Line Only)

An optional backup polling policy defines a directory where a protected database places backup sets without interacting directly with a Recovery Appliance. The backup polling directory must be created on an external file system and made accessible to a Recovery Appliance as an NFS mount point. The polling policy defines the file system path to the storage and how often the Recovery Appliance checks it for new backup sets (not image copies). Specify the polling policy name within a protection policy.

Copyright © 2014, 2026, Oracle and/or its affiliates.

> **Note**
>
> The separate step of creating a backup polling policy is not necessary in Cloud Control, which includes it in the Create Protection Policy page.

Backup polling policies are useful for the following reasons:

- If a Recovery Appliance is offline, then protected databases can continue to send backups to backup polling locations. When a Recovery Appliance comes online, it can poll these locations for backups.
- If the storage network is faster than your Ethernet, and if you configure the polling location in network storage, then protected database backups to the polling location may be faster.
- You can use a polling location when migrating legacy backups to a Recovery Appliance.

Protected databases that use backup polling store backup pieces and archived redo log files in shared storage. The Recovery Appliance periodically retrieves and processes backups from the shared storage.

Prerequisites

You must log in to the metadata database as `RASYS`.

Assumptions

Assume that you want to create a polling policy that meets the following requirements:

- The Recovery Appliance must poll the `/u03/shared/polling1` directory, which is a shared directory accessible by the Recovery Appliance and all protected databases.
- You want the Recovery Appliance to poll the shared directory every 4 hours.
- You want the Recovery Appliance to delete backups from the shared directory after it has processed them.

To create a backup polling policy:

1. Start SQL*Plus or SQL Developer, and then log in to the metadata database as `RASYS`.
2. Run the `DBMS_RA.CREATE_POLLING_POLICY` procedure.

   For example, execute the following PL/SQL anonymous block:

   ```
   BEGIN
     DBMS_RA.CREATE_POLLING_POLICY (
       polling_policy_name   => 'nas_polling1',
       polling_location      => '/u03/shared/polling1',
       polling_frequency     =>  INTERVAL '4' HOUR,
       delete_input          =>  TRUE);
   END;
   ```

3. Optionally, query the recovery catalog to confirm creation of the policy.

   For example, query `RA_POLLING_POLICY` as follows (sample output included):

   ```
   COL POLLING_NAME FORMAT a15
   COL DEST FORMAT a40
   SELECT POLLING_NAME, DEST, DELETE_INPUT,
          TO_CHAR(EXTRACT(DAY FROM FREQUENCY),'fm00')||':'||
          TO_CHAR(EXTRACT(HOUR FROM FREQUENCY),'fm00')||':'||
          TO_CHAR(EXTRACT(MINUTE FROM FREQUENCY),'fm00')||':'||
   ```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
       TO_CHAR(EXTRACT(SECOND FROM FREQUENCY),'fm00')
         AS "DD:HH:MM:SS"
FROM   RA_POLLING_POLICY;

POLLING_NAME    DEST                                    DELET DD:HH:MM:SS
--------------- --------------------------------------- ----- ---------------
NAS_POLLING1    /u03/shared/polling1/                   TRUE  00:04:00:00
```

See Also

- "Backup Polling Policies" for more information about polling
- "CREATE_POLLING_POLICY" for descriptions of procedure arguments

Copyright © 2014, 2026, Oracle and/or its affiliates.

8

Configuring Recovery Appliance for Protected Database Access

This chapter contains the following topics:

- Overview of Protected Database Access
- Basic Tasks for Configuring Protected Database Access
- Creating Virtual Private Catalog Accounts
- Enrolling Protected Databases
- Updating Protected Database Properties
- User Interfaces for Configuring Protected Database Access

8.1 Overview of Protected Database Access

A database is not protected by a Recovery Appliance until it can access the database backups.

Performing necessary configuration so that a protected database can send backups to Recovery Appliance is called enrolling a database. Enrolling is a one-time task that must be performed the first time you set up a protected database to use Recovery Appliance. This task requires configuration on both the Recovery Appliance and the protected database.

The basic enrollment steps are as follows:

1. Adding the database

   The process of adding a database to a Recovery Appliance adds metadata for the database to the Recovery Appliance metadata database, and assigns this database to the specified protection policy. The result of running `DBMS_RA.ADD_DB` is that a non-protected database attains the status of a protected database.

2. Granting access to the database to a Recovery Appliance User Account

   After you create a virtual private catalog account (the Recovery Appliance user) in the metadata database, run `DBMS_RA.GRANT_DB_ACCESS` on the Recovery Appliance to associate this account with the protected database.

3. Registering the database with the virtual private catalog

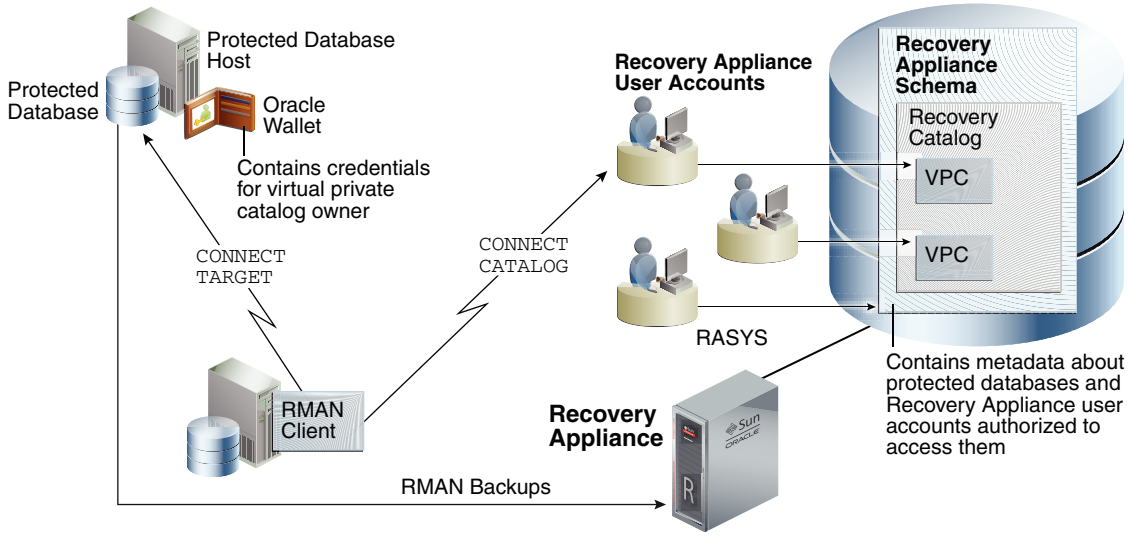
   On the protected database host, create an Oracle wallet, and then add the credentials of the virtual private catalog account. Register the protected database with the recovery catalog using the RMAN `REGISTER DATABASE` command.

> **Note**
>
> If you choose to configure real-time redo transport, then you must execute several SQL statements on the protected database (see *Zero Data Loss Recovery Appliance Protected Database Configuration Guide*).

Copyright © 2014, 2026, Oracle and/or its affiliates.

Figure 8-1 shows an RMAN client connecting to a protected database (`CONNECT TARGET`) and to the virtual private catalog (`CONNECT CATALOG`). For backup and restore operations to be possible, the credentials for the virtual private catalog owner must exist in the Oracle wallet on the protected database host.

Figure 8-1 Protected Database Access

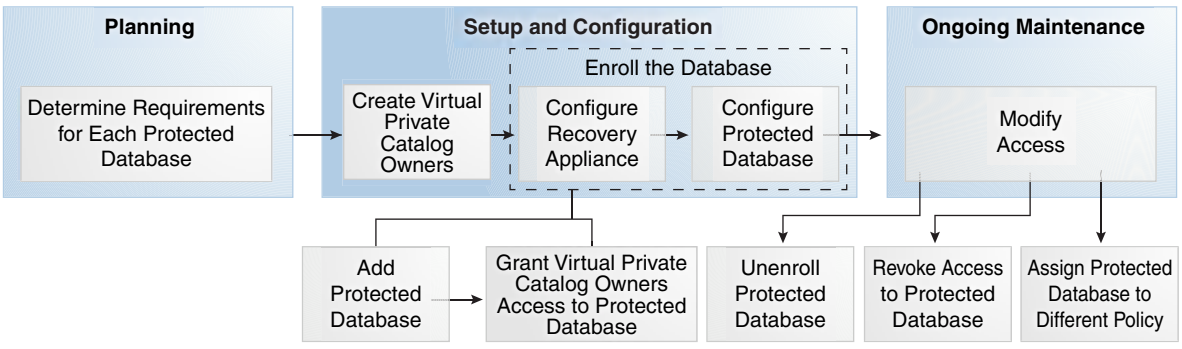


It is possible for a database to store metadata in the Recovery Appliance catalog *without* backing up files to Recovery Appliance. In this case, the databases do not have the status of protected databases, and thus are not enrolled with Recovery Appliance. Future enrolling of such databases is simplified because the virtual private catalog owner already exists, and thus does not need to be created.

8.2 Basic Tasks for Configuring Protected Database Access

This section explains the basic tasks involved in configuring protected database access. Figure 8-2 shows the overall workflow described in Recovery Appliance Workflow , with the configuration tasks on the Recovery Appliance highlighted.

Figure 8-2 Database Access Configuration Tasks in the Recovery Appliance Workflow



Copyright © 2014, 2026, Oracle and/or its affiliates.

Typically, you configure protected database access in the following sequence:

1. During the planning phase, decide which databases will be protected by the Recovery Appliance.

   "Task 4: Determine access requirements for Recovery Appliance" describes this task.

2. During the configuration phase (see "Setup and Configuration for Recovery Appliance"), do the following:

   a. Create virtual private catalog accounts.

      "Creating Virtual Private Catalog Accounts" describes this task.

   b. Enroll the protected database with the Recovery Appliance.

      > **Note**
      >
      > With Cloud Control, you can perform all enrollment steps in a single page *except* registering the database in the recovery catalog.

      "Enrolling Protected Databases" describes this task.

3. During the ongoing maintenance phase (see "Maintenance Tasks for Recovery Appliance"), you can do the following:

   - Update the properties of an existing protected database using `DBMS_RA.UPDATE_DB` (see "Updating Protected Database Properties")
   - Remove metadata for protected databases from the Recovery Appliance using `DBMS_RA.DELETE_DB`
   - Revoke access to a specific protected database from a specific virtual private catalog owner by using `DBMS_RA.REVOKE_DB_ACCESS`

8.3 Creating Virtual Private Catalog Accounts

RMAN must connect to the Recovery Appliance catalog when backing up to a Recovery Appliance. In this step, you create a virtual private catalog user for a specific protected database or set of protected databases.

Assumptions

Assume that you are a Recovery Appliance administrator with the following requirements:

- You want to enroll database `orcld` with a Recovery Appliance.
- You want to create a virtual private catalog account named `ravpc1`. When backing up `orcld`, you plan to run `CONNECT CATALOG` with the `ravpc1` credentials.

To create a virtual private catalog account:

1. Log in to the Recovery Appliance as your named `db_user` with the user type `admin`.
2. Change to the `bin` directory:

   ```
   # cd /opt/oracle.RecoveryAppliance/bin
   ```

3. Run the command to add the new virtual private catalog account.

Copyright © 2014, 2026, Oracle and/or its affiliates.

The following command adds a virtual private catalog account named `ravpc1`:

```
# ./racli add db_user --user_name=ravpc1 --user_type=vpc
```

When prompted, enter the password for the `ravpc1` user.

See Also

- *Oracle Database Security Guide* to learn how to create database user accounts
- *Oracle Database Backup and Recovery User's Guide* to learn about virtual private catalogs

8.4 Enrolling Protected Databases

This section explains how to enroll a protected database using either Cloud Control (recommended) or the `DBMS_RA` command-line interface.

See Also

My Oracle Support Note Doc ID KB439308 for main prerequisites for enrolling a database with Recovery Appliance

- Cloud_Control
- Command_Line

Cloud_Control

To enroll a protected database on the Recovery Appliance with Cloud Control:

1. Using the **Targets** dropdown, select the **Databases** item.

   This opens a screan for **Databases**.

2. From the table of **Databases**, select the database that is to be configured for backup and recovery protection. You may need to login as an administrator for that database.

   This loads a screen with graphical information about the performance of the database, as well as subsequent drop-downs for **Performance**, **Availability**, **Security**, **Schema**, and **Administration**.

3. From the **Availability** drop-down for your chosen database, highlight **Backup & Recovery** and from its flyout select **Configure Backup**.

4. From the **Configure Backup** screen for your chosen database, select from the **Destination** drop-down the option **Recovery Appliance**.

   For the **Host Credentials** field, search for a **Named** user who has database host credentials.

5. In the subsequent screen for **Configure Backup** for your chosen database:

Copyright © 2014, 2026, Oracle and/or its affiliates.

- From the **Recovery Appliance** drop-down, select the destination recovery appliance.
- From the **Virtual Private Catalog User** drop-down, select the approriate VPC user.
- Establish other options for the database.

  If the protection policy has auto tune enabled, the reserved space does not need to be specified.

If the Recovery Appliance only supports TCP or TCPS-only, the **Protocol** cannot be changed. Only if the Recovery Appliance is in dual mode does the drop-down provide options for TCP and TCPS.

Enable Real-Time Redo switch is only permissible if the database is not a Data Guard database and if the database is in archive log mode.

The **Wallet** field shows the path to the wallet if it exists, and the field is disabled. If no wallet exists, you can provide a location or use the recommended location for the wallet.

For Oracle Databases prior to DB 23, the user can choose to **Install Backup Module** if it is needed.

6. When finished with the **Configure Backup** settings for your chosen database, **Submit** them.

Command_Line

To enroll a protected database on a Recovery Appliance with PL/SQL:

When enrolling databases using the `DBMS_RA` command-line interface, you must perform the following tasks:

1. Add protection database metadata to the Recovery Appliance using `DBMS_RA`
2. Grant database access to a Recovery Appliance account using `DBMS_RA`.
3. Configuring the protected database for access (see *Zero Data Loss Recovery Appliance Protected Database Configuration Guide*)

Add Metadata for the Database

For a database to be protected, you must add metadata for this database to the Recovery Appliance using `DBMS_RA.ADD_DB`. This procedure requires you to specify an existing protection policy and the amount of reserved space for the database.

You must log in to the Recovery Appliance with the `RASYS` account or with a named `db_user` with `user_type=admin`.

The following examples assumes the following requirements:

- You want to make `orcld` a protected database.
- You want to add this database to the existing `bronze` protection policy, and provide it with 200 GB of reserved space.

1. With SQL*Plus or SQL Developer, connect to the Recovery Appliance metadata database as `RASYS`.
2. Use the `ADD_DB` procedure to add database metadata to the Recovery Appliance and assign a protection policy.

   For example, the following anonymous block adds database `orcld`:

```
BEGIN
  DBMS_RA.ADD_DB (
    db_unique_name         => 'orcld',
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
    protection_policy_name => 'bronze',
    reserved_space         => '200G');
END;
```

3. Optionally, query the recovery catalog to see information about the newly added database.

   For example, execute the following query to show details about `orcld` (sample output included):

```
COLUMN PROT_DB FORMAT a10
COLUMN POLICY_NAME FORMAT a11
SELECT DB_UNIQUE_NAME AS PROT_DB, DB_KEY, DBID, POLICY_NAME
FROM   RA_DATABASE
WHERE  DB_UNIQUE_NAME = 'ORCLD';

PROT_DB        DB_KEY       DBID POLICY_NAME
---------- ---------- ---------- -----------
ORCLD             301 3210984255 BRONZE
```

> **Note**
>
> In an Oracle Data Guard environment, add the `db_unique_name` of whichever database (primary or standby) that you registered with the Recovery Appliance catalog.

> **See Also**
>
> "ADD_DB"

Granting Access

You must grant the necessary privileges to a Recovery Appliance user account—which is also a virtual private catalog account—so that protected databases that authenticate with this account can perform backup and restore operations. The `DBMS_RA.GRANT_DB_ACCESS` procedure associates a protected database with a virtual private catalog.

- You must log in to the Recovery Appliance with the `RASYS` account or with a named `db_user` with `user_type=admin`..
- The the Recovery Appliance user account specified in `DBMS_RA.GRANT_DB_ACCESS` must exist.
- You must have already added the protected database named `orcld`.

For this example, assume that you want to enable RMAN to `CONNECT CATALOG` as `ravpc1` when backing up protected database `orcld`.

1. With SQL*Plus or SQL Developer, connect to the Recovery Appliance database as `RASYS`.
2. Run the `GRANT_DB_ACCESS` procedure to grant backup and restore privileges on the database for the user.

   The following PL/SQL anonymous block grants access to protected database `orcld` to virtual private catalog account `ravpc1`:

```
BEGIN
   DBMS_RA.GRANT_DB_ACCESS (
     db_unique_name =>  'orcld',
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
     username       =>  'ravpc1');
END;
```

3. Optionally, query the recovery catalog to see information about the database access.

   For example, execute the following query to show details about `orcld` and catalog owner `ravpc1` (sample output included):

```
COLUMN PROT_DB FORMAT a10
COLUMN POLICY_NAME FORMAT a11
COLUMN USERNAME FORMAT a15
COLUMN DB_KEY FORMAT 999999
SELECT d.DB_UNIQUE_NAME AS PROT_DB, d.DB_KEY,
       d.DBID, d.POLICY_NAME, a.USERNAME
FROM   RA_DATABASE d, RA_DB_ACCESS a
WHERE  d.DB_UNIQUE_NAME = 'ORCLD'
AND    a.DB_KEY = d.DB_KEY;

PROT_DB     DB_KEY       DBID POLICY_NAME USERNAME
---------- ------- ---------- ----------- ---------------
ORCLD          301 3210984255 BRONZE      RAVPC1
```

4. Send the virtual private catalog user name and password to the DBA for each protected database that must authenticate using this account.
5. To complete the enrollment procedure, see *Zero Data Loss Recovery Appliance Protected Database Configuration Guide*.

> **See Also**
>
> "GRANT_DB_ACCESS"

8.5 Updating Protected Database Properties

This section explains how to update protected database properties using either Cloud Control (recommended) or the `DBMS_RA` command-line interface.

- Cloud_Control
- Command_Line

Cloud_Control

To update a protected database properties with Cloud Control:

Assume that you have the following business requirements:

- You want to change the protection policy for protected database `ORCL11` from `GOLD` to `BRONZE`.
- You want change the reserved space from 6355 GB to 7000 GB.
- You want to change the Recovery Appliance user account associated with this protected database from `rauser11` to `rauser12`.

Copyright © 2014, 2026, Oracle and/or its affiliates.

1. Access the Protected Databases page, as described in "Accessing the Recovery Appliance Home Page".
2. Click **Edit**.

   The Edit Protected Databases page appears.
3. Change the desired attributes of the protected database, and then click **OK**:
   - In the **Protection Policy** section, select the row for the policy named `BRONZE`.

     For example, select **All**.
   - In the **Reserved Space** field, enter the new minimum amount of disk space to be reserved for this protected database.

     For example, enter `7000`, and then select **GB** for the units.
   - In the **Recovery Appliance User** section, enter the credentials for the database user `rauser12`.

   The newly updated database appears in the table of protected databases.

Command_Line

To update a protected database properties with DBMS_RA:

Assume that you have the following business requirements:

- You want to change the protection policy for protected database `zdlrac` from `silver` to `bronze`.

1. Start SQL*Plus or SQL Developer, and then log in to the metadata database as `RASYS` or as a named `db_user` with `user_type=admin`.
2. Query the existing protection policies.

   For example, execute the following query (sample output included):

```
COL POLICY_NAME FORMAT a11
COL DESCRIPTION FORMAT a35
SELECT POLICY_NAME, DESCRIPTION,
       TO_CHAR(EXTRACT(DAY FROM RECOVERY_WINDOW_GOAL),'fm00')||':'||
       TO_CHAR(EXTRACT(HOUR FROM RECOVERY_WINDOW_GOAL),'fm00')||':'||
       TO_CHAR(EXTRACT(MINUTE FROM RECOVERY_WINDOW_GOAL),'fm00')||':'||
       TO_CHAR(EXTRACT(SECOND FROM RECOVERY_WINDOW_GOAL),'fm00')
         AS "DD:HH:MM:SS"
FROM   RA_PROTECTION_POLICY;

POLICY_NAME DESCRIPTION                         DD:HH:MM:SS
----------- ----------------------------------- --------------
BRONZE      For protected dbs in bronze tier    01:00:00:00
SILVER      For protected dbs in silver tier    07:00:00:00
GOLD        For protected dbs in gold tier      14:00:00:00
```

3. Determine which protected databases are associated with which protection policies.

   For example, execute the following query (sample output included):

```
SELECT d.DB_UNIQUE_NAME, d.POLICY_NAME
FROM   RA_PROTECTION_POLICY p, RA_DATABASE d
WHERE  p.policy_name=d.policy_name
ORDER BY d.DB_UNIQUE_NAME;

DB_UNIQUE_NAME                  POLICY_NAME
------------------------------- -----------
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
ZDLRA                          BRONZE
ZDLRAC                         SILVER
.
.
.
```

4. Run the `DBMS_RA.UPDATE_DB` procedure to associate a database with a new policy.

   For example, execute the following PL/SQL anonymous block to associate the database named `zdlrac`, which has `silver` as its current policy, with the protection policy named `bronze`:

```
BEGIN
 DBMS_RA.UPDATE_DB(
   db_unique_name             => 'zdlrac',
   protection_policy_name     => 'bronze');
END;
```

5. Optionally, confirm that the database is associated with the correct policy.

   For example, execute the following query (sample output included):

```
SELECT d.DB_UNIQUE_NAME, d.POLICY_NAME
FROM   RA_PROTECTION_POLICY p, RA_DATABASE d
WHERE  p.POLICY_NAME=d.POLICY_NAME
ORDER BY d.DB_UNIQUE_NAME;

DB_UNIQUE_NAME                 POLICY_NAME
------------------------------ -----------
ZDLRA                          BRONZE
ZDLRAC                         BRONZE
.
.
.
```

> **See Also**
>
> "UPDATE_DB"

8.6 User Interfaces for Configuring Protected Database Access

This section contains the following topics:

- DBMS_RA Procedures Relating to Protected Database Access
- Recovery Catalog Views for Protected Database Access

8.6.1 DBMS_RA Procedures Relating to Protected Database Access

You can use the `DBMS_RA` package to configure protected database access. Table 8-1 describes the principal program units relating to protected databases.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 8-1 DBMS_RA Protected Database Access Procedures

Program Unit	Description
ADD_DB	Adds metadata for the specified database to Recovery Appliance, and assigns a protection policy to the database. Note that you must set the `reserved_space` parameter unless the protection policy is using `autotune_reserved_space`.
DELETE_DB	Removes metadata for the specified database from Recovery Appliance. All metadata and backups of this database are deleted, from both disk and SBT.
SUSPEND_DB	Removes metadata for the specified database from Recovery Appliance. All metadata and backups of this database are deleted, from both disk and SBT. Backups on tape, in the cloud, or replicated to other Recovery Appliances are not affected. While a database is suspended, it will not accept backups. The database must be resumed before it can return to normal operation.
RESUME_DB	Restores a suspended database to normal operation. Only suspended databases may be resumed. Suspended databases must be resumed before they can be backed up.
UPDATE_DB	Changes the attributes that are assigned to the specified protected database.
GRANT_DB_ACCESS	Grants Recovery Appliance privileges to a user for a specified database.
REVOKE_DB_ACCESS	Revokes Recovery Appliance privileges from a user for a specified database.

See Also

DBMS_RA Package Reference

8.6.2 Recovery Catalog Views for Protected Database Access

You can monitor database access using the Recovery Appliance catalog views. Table 8-2 summarizes the most relevant views.

Table 8-2 Recovery Catalog Views for Protected Database Access

View	Description
RA_DATABASE	This view describes databases protected by this Recovery Appliance.
RA_DB_ACCESS	This view describes the user account that can access specific protected databases.

See Also

Recovery Appliance View Reference

Copyright © 2014, 2026, Oracle and/or its affiliates.

9

Copying Backups to Tape with Recovery Appliance

This chapter explains how to copy completed backups to tape to ensure optimal utilization of space and storage resources on Zero Data Loss Recovery Appliance.

This chapter contains the following topics:

- About Copying Backups to Tape with Recovery Appliance
- Creating Tape Backup Job Components
- Managing Tape Backup Job Components
- Creating a Tape Backup Job
- Managing a Tape Backup Job
- Scheduling a Tape Backup Job
- Pausing and Resuming Tape Backup Operations
- Viewing the Status of Tape Backup Operations

9.1 About Copying Backups to Tape with Recovery Appliance

This section contains the following topics:

- Purpose of Copying Backups to Tape with Recovery Appliance
- Overview of Copying Backups to Tape with Recovery Appliance
- User Interfaces for Recovery Appliance
- Basic Tasks for Copying Backups to Tape with Recovery Appliance

9.1.1 Purpose of Copying Backups to Tape with Recovery Appliance

A robust backup strategy protects data against intentional attacks, unintentional user errors (such as file deletions), and software or hardware malfunctions. Tape libraries provide effective protection against these possibilities.

The advantages of the Recovery Appliance tape solution are as follows:

- Tape backups are ideal for long-term storage. Tapes are portable and easy to store for lengthy periods of time.
- All tape backup operations are performed by the Recovery Appliance, with no performance load on the protected database host.
- Tape backups are optimized. Recovery Appliance intelligently gathers the necessary blocks to create a virtual, full backup or incremental backup for tape. Although Recovery Appliance backups are incremental forever, you can create a flexible backup strategy to tape, such as weekly full and daily incremental or just daily full backups.

Copyright © 2014, 2026, Oracle and/or its affiliates.

- Oracle Secure Backup is pre-installed, eliminating the need for third-party media managers.
- Tape drives and tape libraries function more efficiently because Recovery Appliance is a single large centralized system with complete control over them. In other tape solutions, hundreds or thousands of databases can compete for tape resources in an uncoordinated manner.

Note

Oracle Secure Backup (OSB) 19.1 desupport was announced on May 1, 2026 with premier support end date on September 30, 2027. See My Oracle Support note PNEWS3035 for more information. Please refer to Lifetime Support Policy: Oracle Technology Products for updated support timeframes.

Oracle recommends that database backups to cloud take advantage of Oracle Database Zero Data Loss Cloud Protect, which uses Oracle Zero Data Loss Autonomous Recovery Service in OCI. Cloud Protect offers an efficient incremental forever backup strategy, real-time transaction protection, logically air-gapped immutable backups, and fast, point-in-time recovery.

For customers who require database backups to Amazon S3, Oracle Database Cloud Backup for Amazon S3 may be used. This offering is similar to Oracle Secure Backup Cloud Module in that it is fully integrated with Recovery Manager (RMAN) and enables you to back up your Oracle database to Amazon S3. You can exchange your Oracle Secure Backup license for an Oracle Database Cloud Backup for Amazon S3 license on a 1:1 basis. See Oracle Database Cloud Backup for Amazon S3 for more information.

As with Oracle Secure Backup Cloud Module, Oracle Database Cloud Backup for Amazon S3 can be used free of charge on Oracle Zero Data Loss Recovery Appliance (ZDLRA) for the sole purpose of backing up data to and restoring data from S3-compatible storage devices that are in the same campus as ZDLRA.

See Also

"Autonomous Tape Archival"

9.1.2 Overview of Copying Backups to Tape with Recovery Appliance

This section contains the following topics:

- About Tape Operations on Recovery Appliance
- Grouping Backup Pieces
- Recovery Appliance Components for Managing Tape Operations
- Backup Retention on Tape
- About Pausing and Resuming Tape Backup Operations

Copyright © 2014, 2026, Oracle and/or its affiliates.

9.1.2.1 About Tape Operations on Recovery Appliance

All backups that Recovery Appliance receives from protected databases are always first stored on disk. The Recovery Appliance can then optionally copy these backups to tape. All copying to tape is automated, policy-driven, and scheduled.

A protection policy defines desired recovery windows for backups stored on tape. Recovery windows are expressed as time intervals, such as 30 days. Backups are retained on tape long enough for a recovery to be possible at any time within this interval, counting backward from the current time.

You can copy Recovery Appliance backups from disk to tape. To perform this task, you must create a tape backup job that defines the properties of the copy operation, such as the media manager library and attribute set that will manage this job, the protection policy or the database for which the backups need to be copied, and so on. After you have defined the job properties, you must schedule this job to run.

Note

- Only backups that have not already been copied to tape are processed in a tape backup operation for each tape backup job template with which the backup is associated. Thus, a tape backup operation on the same backup after the initial tape copy has no effect. In addition, only the most recent backup is copied to tape when the tape backup operation runs.

  If you require more than one copy of the same backup, such as to a different media family on tape, use the `COPIES` parameter of the template or create a separate tape backup job template for the additional copy.
- Virtual full backups copied to tape or cloud use RMAN `FILESPERSET=1` setting. Incremental backups copied to tape use the `FILESPERSET` setting as specified in the RMAN incremental backup command to the Recovery Appliance.
- Backup pieces, such as archive logs, are grouped together and copied as a single piece. These backup pieces are larger on cloud or tape storage. This feature is disabled by default and can group a maximum of 64 archived logs per backup piece that is copied to cloud or tape. The effects of inter-job latencies are reduced when fewer individual pieces are transmitted.
- Long-term archival backups that were created with the `KEEP` option of the `BACKUP` command are never automatically copied to tape. You must manually copy them using the COPY_BACKUP or MOVE_BACKUP procedure.

  See My Oracle Support Note Doc ID KB475182 to learn how to create archival backups for long term retention on the Recovery Appliance

During a restore, Recovery Appliance transparently retrieves the backup from tape.

Recovery Appliance writes backups to tape in formats supported by RMAN. If a protected database has the required media management software (for example, Oracle Secure Backup), then it can directly restore backups written to tape by the Recovery Appliance.

Copyright © 2014, 2026, Oracle and/or its affiliates.

> **Note**
>
> When the backups are created as *space-efficient backups* (using compression and TDE encryption), their restoration requires the Recovery Appliance, because the database communicates with the appliance to process space-efficient backups. As such, these backups cannot be restored directly from the media.

9.1.2.2 Grouping Backup Pieces

The performance of copy-to-tape and archive-to-cloud is improved by grouping archived logs from protected databases' real-time redo into fewer number of backup sets.

Protected databases can achieve real-time protection by enabling real-time redo transport to the Recovery Appliance. Each received redo log on the appliance is compressed and written to the storage location as an individual archived log backup. These log backups can be archived to tape to support fulls and incremental backups that are archived for long-term retention needs.

Inter-job latencies can happen between writing each backup piece during copy-to-tape operations. When the number of backup pieces is high, this pause constitutes a large percentage of the time the tape drive is unavailable. This means five (5) 10GB pieces will go to tape more quickly than fifty (50) 1GB pieces.

Recovery Appliance addresses inter-job latency by grouping the archived log backup pieces together and copying them as a single backup piece. Therefore this results in larger backup pieces on tape storage than previous releases. This feature is enabled by default. `DMBS_RA CONFIG` has the parameter `group_log_max_count` for setting the maximum archived logs per backup piece that is copied to tape; its default is 1. The `group_log_backup_size_gb` parameter is used to limit the size of these larger backup pieces; its default is 256 GB.

> **Note**
>
> Storing backups in the cloud does not have the latency of tape devices. The number of streams that can be assigned in the cloud is not limited by the (tape) device. Grouping archived log files is recommended for tape operations, but not necessarily cloud.

9.1.2.3 Recovery Appliance Components for Managing Tape Operations

Every database in a Recovery Appliance setup is associated with a protection policy that specifies the parameters for backup storage and recovery window goal. To manage and control tape operations, you must create a job that uses the properties defined by the selected protection policy, media manager library, and attribute set to copy backups to tape. Oracle Secure Backup and its components (media manager library and attribute sets) are preconfigured with Recovery Appliance.

Copyright © 2014, 2026, Oracle and/or its affiliates.

> **Note**
>
> Oracle Secure Backup (OSB) 19.1 desupport was announced on May 1, 2026 with premier support end date on September 30, 2027. See My Oracle Support note PNEWS3035 for more information. Please refer to Lifetime Support Policy: Oracle Technology Products for updated support timeframes.
>
> Oracle recommends that database backups to cloud take advantage of Oracle Database Zero Data Loss Cloud Protect, which uses Oracle Zero Data Loss Autonomous Recovery Service in OCI. Cloud Protect offers an efficient incremental forever backup strategy, real-time transaction protection, logically air-gapped immutable backups, and fast, point-in-time recovery.
>
> For customers who require database backups to Amazon S3, Oracle Database Cloud Backup for Amazon S3 may be used. This offering is similar to Oracle Secure Backup Cloud Module in that it is fully integrated with Recovery Manager (RMAN) and enables you to back up your Oracle database to Amazon S3. You can exchange your Oracle Secure Backup license for an Oracle Database Cloud Backup for Amazon S3 license on a 1:1 basis. See Oracle Database Cloud Backup for Amazon S3 for more information.
>
> As with Oracle Secure Backup Cloud Module, Oracle Database Cloud Backup for Amazon S3 can be used free of charge on Oracle Zero Data Loss Recovery Appliance (ZDLRA) for the sole purpose of backing up data to and restoring data from S3-compatible storage devices that are in the same campus as ZDLRA.

Table 9-1 summarizes the roles of the Recovery Appliance objects for managing tape operations.

Table 9-1 Recovery Appliance Objects for Copying Backups to Tape

Cloud Control Object Name	Command-line Object Name	Description
Protection policy	Protection policy	Among other attributes, defines the recovery window. This recovery window is applied to all protected databases assigned to the protection policy.
Media manager library	SBT library	Describes a media management software library installed on Recovery Appliance.
Media manager attribute set	SBT attribute set	Contains a collection of attributes that control the copy operation. One attribute specifies the library to be used in the copy operation. Other attributes are optional and can include channel parameters, media management software library–specific commands, and a media pool identifier. You can define multiple attribute sets, but only one attribute set is associated with a given copy job.
Copy-to-tape job template	SBT job template	Defines the properties of backups to be copied to tape and specifies an attribute set to control the copy operation. Typically, multiple job templates are defined.

Copyright © 2014, 2026, Oracle and/or its affiliates.

> **Note**
>
> The Oracle Enterprise Manager Cloud Control (Cloud Control) object name and the command-line name in Table 9-1 refer to the same tape backup objects, with their respective interface terms.

> **See Also**
>
> "Basic Tasks for Copying Backups to Tape with Recovery Appliance" for more information on how to create a tape backup job using these components

9.1.2.4 Backup Retention on Tape

You can control the length of time that backup copies are retained on tape by specifying a recovery window. A recovery window defines how long the Recovery Appliance maintains tape backups in its catalog for recovery purposes. The recovery window is expressed as an interval, in values of hours, days, weeks, or months. Backups are retained long enough to guarantee that a recovery is possible to any point in time within this interval, counting backward from the current time.

> **Note**
>
> Recovery windows directly apply only to full or level 0 data file and control file backups.

Recovery Appliance does not purge tape backups. Instead, it informs the media manager which pieces are no longer needed for RMAN retention. With Oracle Secure Backup as the media manager, it does not explicitly delete these files, it updates its catalog. After all files on a given tape are no longer needed, Oracle Secure Backup considers the tape for reuse.

You set the recovery window for a backup by providing a value for it in the protection policy. If this attribute is `NULL`, then Recovery Appliance never purges the backup from tape.

> **See Also**
>
> - "CREATE_SBT_JOB_TEMPLATE"
> - "Creating a Protection Policy" for more information on setting recovery window goals using Cloud Control and the command line.
> - *Oracle Database Backup and Recovery User's Guide* for a thorough discussion of recovery windows

9.1.2.5 About Pausing and Resuming Tape Backup Operations

You might want to pause the copying of backups to tape for these reasons:

- To investigate previous backup copy failures

Copyright © 2014, 2026, Oracle and/or its affiliates.

- To perform maintenance operations on tape devices

You pause tape backup operations for a specific media management software library by pausing its corresponding media manager library.

When you pause a media manager library, in-progress copies of backup pieces are allowed to complete, while backup pieces that were queued for copy but not yet copied are held until you resume the library. Pausing a library suspends future scheduled runs of tape backup jobs that reference the media manager library.

> **Note**
>
> Tape backup jobs reference media manager libraries indirectly through their assigned media manager attribute sets.

> **See Also**
>
> - "Pausing and Resuming Tape Backup Operations "

9.1.2.6 About Using Oracle Secure Backup with Recovery Appliance

Oracle Secure Backup is a media manager that provides reliable, centralized tape management by protecting file-system data and Oracle AI Database files for multiple environments. Oracle Secure Backup is the tape management component for Recovery Appliance. It is installed with its components on the Recovery Appliance during its configuration.

Copyright © 2014, 2026, Oracle and/or its affiliates.

> **Note**
>
> Oracle Secure Backup (OSB) 19.1 desupport was announced on May 1, 2026 with premier support end date on September 30, 2027. See My Oracle Support note PNEWS3035 for more information. Please refer to Lifetime Support Policy: Oracle Technology Products for updated support timeframes.
>
> Oracle recommends that database backups to cloud take advantage of Oracle Database Zero Data Loss Cloud Protect, which uses Oracle Zero Data Loss Autonomous Recovery Service in OCI. Cloud Protect offers an efficient incremental forever backup strategy, real-time transaction protection, logically air-gapped immutable backups, and fast, point-in-time recovery.
>
> For customers who require database backups to Amazon S3, Oracle Database Cloud Backup for Amazon S3 may be used. This offering is similar to Oracle Secure Backup Cloud Module in that it is fully integrated with Recovery Manager (RMAN) and enables you to back up your Oracle database to Amazon S3. You can exchange your Oracle Secure Backup license for an Oracle Database Cloud Backup for Amazon S3 license on a 1:1 basis. See Oracle Database Cloud Backup for Amazon S3 for more information.
>
> As with Oracle Secure Backup Cloud Module, Oracle Database Cloud Backup for Amazon S3 can be used free of charge on Oracle Zero Data Loss Recovery Appliance (ZDLRA) for the sole purpose of backing up data to and restoring data from S3-compatible storage devices that are in the same campus as ZDLRA.

Preconfigured Oracle Secure Backup components include the following:

Media Manager Library
During its installation, while Recovery Appliance is being configured, Oracle Secure Backup creates a media manager library with default parameters, such as the following:

- Library name (`ROBOT0`)
- Maximum number of accessible tape drives
- Number of restore drives
- Media manager location

Apart from the name, other media manager library parameters can be modified. This library manages the tape backup operations associated with it, based on the parameters set.

Media Manager Attribute Sets
Along with a media manager library, Oracle Secure Backup also comes installed with attribute sets for all tape drives that the default media manager library accesses.
These attribute sets have default values for parameters like the media pool number and streams required for the copy operation. These and the media manager vendor parameters and commands can be modified. The default attribute sets are named `DRIVE_COUNT_1`, `DRIVE_COUNT_2`, `DRIVE_COUNT_3`, and so on for the number of tape drives accessed by the media manager library.

Copyright © 2014, 2026, Oracle and/or its affiliates.

See Also

- "Accessing the Oracle Secure Backup Domain Using Cloud Control" for more information on how to access the Oracle Secure Backup domain using Cloud Control
- "Creating Tape Backup Job Components" for more information on how to create a media manager library and attribute sets for third-party media managers
- "Managing Tape Backup Job Components" for more information on how to edit and control existing media manager components

9.1.3 User Interfaces for Recovery Appliance

You can manage and perform tape backup job operations by using either Cloud Control or the Recovery Appliance command-line options.

This section contains the following topics:

- Accessing Recovery Appliance in Cloud Control
- Accessing Recovery Appliance Using DBMS_RA

9.1.3.1 Accessing Recovery Appliance in Cloud Control

To access Recovery Appliance using Cloud Control, complete the steps listed in "Accessing the Recovery Appliance Home Page".

9.1.3.2 Accessing Recovery Appliance Using DBMS_RA

This section contains the following topics:

- DBMS_RA Procedures for Tape Backup Operations
- Recovery Catalog Views for Tape Operations

9.1.3.2.1 DBMS_RA Procedures for Tape Backup Operations

This section lists the Recovery Appliance `DBMS_RA` procedures that are associated with SBT job operations. They are applicable for tape, cloud, and archive operations.

Table 9-2 DBMS_RA Procedures Associated with Tape/Cloud/Archive Backup Operations

SBT Object	Procedures
SBT Job	• CREATE_SBT_JOB_TEMPLATE • UPDATE_SBT_JOB_TEMPLATE • DELETE_SBT_JOB_TEMPLATE
SBT Library	• CREATE_SBT_LIBRARY • UPDATE_SBT_LIBRARY • PAUSE_SBT_LIBRARY • RESUME_SBT_LIBRARY • DELETE_SBT_LIBRARY

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 9-2 (Cont.) DBMS_RA Procedures Associated with Tape/Cloud/Archive Backup Operations

SBT Object	Procedures
SBT Attribute Set	• CREATE_SBT_ATTRIBUTE_SET • UPDATE_SBT_ATTRIBUTE_SET • DELETE_SBT_ATTRIBUTE_SET
Protection Policy	• CREATE_PROTECTION_POLICY • UPDATE_PROTECTION_POLICY • DELETE_PROTECTION_POLICY
Backup	• QUEUE_SBT_BACKUP_TASK • COPY_BACKUP • MOVE_BACKUP

See Also

"DBMS_RA Package Reference" for more information on other Recovery Appliance `DBMS_RA` procedures

9.1.3.2.2 Recovery Catalog Views for Tape Operations

This section lists the Recovery Appliance recovery catalog views that are associated with SBT job operations:

- RA_SBT_JOB
- RA_SBT_LIBRARY
- RA_SBT_ATTRIBUTE_SET
- RA_PROTECTION_POLICY
- RA_EM_SBT_JOB_TEMPLATE

See Also

Recovery Appliance View Reference for more information on other Recovery Appliance data dictionary views

9.1.4 Basic Tasks for Copying Backups to Tape with Recovery Appliance

This section lists the high level essential steps to copy database backups to tape using Recovery Appliance.

See Also

"About Copying Backups to Tape with Recovery Appliance"

Copyright © 2014, 2026, Oracle and/or its affiliates.

To copy backups to tape using Recovery Appliance:

1. Create a media manager library for your media management software to manage all your tape backup jobs by adding parameters that apply to a set of jobs.

   Recovery Appliance uses Oracle Secure Backup as its media management software. During its setup, Recovery Appliance installs Oracle Secure Backup with a preconfigured media manager library and attribute sets.

   > **Note**
   >
   > Oracle Secure Backup (OSB) 19.1 desupport was announced on May 1, 2026 with premier support end date on September 30, 2027. See My Oracle Support note PNEWS3035 for more information. Please refer to Lifetime Support Policy: Oracle Technology Products for updated support timeframes.
   >
   > Oracle recommends that database backups to cloud take advantage of Oracle Database Zero Data Loss Cloud Protect, which uses Oracle Zero Data Loss Autonomous Recovery Service in OCI. Cloud Protect offers an efficient incremental forever backup strategy, real-time transaction protection, logically air-gapped immutable backups, and fast, point-in-time recovery.
   >
   > For customers who require database backups to Amazon S3, Oracle Database Cloud Backup for Amazon S3 may be used. This offering is similar to Oracle Secure Backup Cloud Module in that it is fully integrated with Recovery Manager (RMAN) and enables you to back up your Oracle database to Amazon S3. You can exchange your Oracle Secure Backup license for an Oracle Database Cloud Backup for Amazon S3 license on a 1:1 basis. See Oracle Database Cloud Backup for Amazon S3 for more information.
   >
   > As with Oracle Secure Backup Cloud Module, Oracle Database Cloud Backup for Amazon S3 can be used free of charge on Oracle Zero Data Loss Recovery Appliance (ZDLRA) for the sole purpose of backing up data to and restoring data from S3-compatible storage devices that are in the same campus as ZDLRA.

   > **See Also**
   >
   > "Creating a Media Manager Library" for more information on how to create additional media manager libraries

2. Create a media manager attribute set that helps you control your tape backup jobs further by adding more job-specific parameters and commands for your media manager software.

   Tape backup jobs use a combination of parameters specified at the media manager library level and the attribute set level while performing the copy operation. Media manager libraries define parameters that apply to a set of jobs while attribute sets help further define tape backup settings for specified jobs.

   Oracle Secure Backup also configures default attribute sets for all drives that are a part of the default media manager library.

Copyright © 2014, 2026, Oracle and/or its affiliates.

> **See Also**
>
> "Creating an Attribute Set" for more information on creating additional attribute sets

3. Create a tape backup job.

   The job definition includes job properties such as the media manager library and attribute set associated with this job, the type of backups that need to be copied to tape, the run-time window for this job, and so on.

   You can also schedule this job to run at a specified time according to your task requirements.

> **See Also**
>
> "Creating a Tape Backup Job" for more information on creating tape backup jobs

4. (Optional) If required, pause or resume a media manager library or a tape backup job.

> **See Also**
>
> "Pausing and Resuming Tape Backup Operations "

5. View the status of all media manager libraries and tape backup operations to check for errors.

> **See Also**
>
> "Viewing the Status of Tape Backup Operations"

You may want to create additional media families for refining backup properties or to schedule vaulting to manage your tape. Use the Recovery Appliance Oracle Secure Backup domain to complete these tasks.

> **See Also**
>
> - "Accessing the Oracle Secure Backup Domain Using Cloud Control"
> - *Oracle Secure Backup Administrator's Guide* for more information about configuring media families
> - *Oracle Secure Backup Administrator's Guide* for more information on vaulting

9.1.4.1 Accessing the Oracle Secure Backup Domain Using Cloud Control

Use Cloud Control to access the Oracle Secure Backup domain. You can use this domain to manage (if required) the existing Oracle Secure Backup configurations set for the selected Recovery Appliance environment.

Copyright © 2014, 2026, Oracle and/or its affiliates.

To access the Oracle Secure Backup domain using Cloud Control:

1. From the Cloud Control Home page, select **Targets**.
2. From the Targets Menu, select **All Targets**.

   The All Targets page appears.
3. On the All Targets page, under the Refine Search Menu, select **Databases** as the Target Type.
4. Under the Databases section, select **Oracle Secure Backup Domain**.

   A list of all Oracle Secure Backup Domains for all existing Recovery Appliance targets appears.
5. From the list of targets, click the target for which you want to access the Oracle Secure Backup domain.

   The Oracle Secure Backup domain for the selected Recovery Appliance appears.

> **See Also**
>
> "About Using Oracle Secure Backup with Recovery Appliance"

9.2 Creating Tape Backup Job Components

To successfully create a tape backup job, you must first ensure that its components exist. A media manager library and its attribute sets are essential for a tape backup operation. These components define a combination of parameters for a tape backup job and help categorize these jobs when storing them on tape.

> **See Also**
>
> "Recovery Appliance Components for Managing Tape Operations" for more information on the role of a media manager library and its attribute sets

This section contains the following topics:

- Creating a Media Manager Library
- Creating an Attribute Set

9.2.1 Creating a Media Manager Library

A media manager library sets the properties for a tape backup job by defining parameters like the number of tape drives it can access. Optional advanced parameters include specifying the number of required restore drives and media manager parameters.

Recovery Appliance comes with Oracle Secure Backup as its preconfigured media manager. During the Recovery Appliance configuration, a media manager library is also configured for Oracle Secure Backup, typically named `ROBOT0`. It is recommended that you only use the preconfigured media manager objects. The media manager (Oracle Secure Backup, in this case) must have only a single media manager library as more than one library object will result

Copyright © 2014, 2026, Oracle and/or its affiliates.

in a conflict between the tape backup jobs created and the media manager resources handling these jobs. Currently, you cannot install Oracle Secure Backup in the Recovery Appliance in `client only`.

> **Note**
>
> Oracle Secure Backup (OSB) 19.1 desupport was announced on May 1, 2026 with premier support end date on September 30, 2027. See My Oracle Support note PNEWS3035 for more information. Please refer to Lifetime Support Policy: Oracle Technology Products for updated support timeframes.
>
> Oracle recommends that database backups to cloud take advantage of Oracle Database Zero Data Loss Cloud Protect, which uses Oracle Zero Data Loss Autonomous Recovery Service in OCI. Cloud Protect offers an efficient incremental forever backup strategy, real-time transaction protection, logically air-gapped immutable backups, and fast, point-in-time recovery.
>
> For customers who require database backups to Amazon S3, Oracle Database Cloud Backup for Amazon S3 may be used. This offering is similar to Oracle Secure Backup Cloud Module in that it is fully integrated with Recovery Manager (RMAN) and enables you to back up your Oracle database to Amazon S3. You can exchange your Oracle Secure Backup license for an Oracle Database Cloud Backup for Amazon S3 license on a 1:1 basis. See Oracle Database Cloud Backup for Amazon S3 for more information.
>
> As with Oracle Secure Backup Cloud Module, Oracle Database Cloud Backup for Amazon S3 can be used free of charge on Oracle Zero Data Loss Recovery Appliance (ZDLRA) for the sole purpose of backing up data to and restoring data from S3-compatible storage devices that are in the same campus as ZDLRA.

If you are using third-party media management software, you must install its backup agent on the Recovery Appliance compute servers. To schedule a tape backup job using the third-party product, you must create a new media manager library and add RMAN parameters applicable for that media manager for backups over LAN to tape devices attached to the backup application's media servers. In this scenario, you will not be able to use the media manager components preconfigured for Oracle Secure Backup and cannot directly attach tape devices to the Recovery Appliance.

This section describes the steps to create an additional media manager library for a third-party media manager.

- Cloud_Control
- Command_Line

Cloud_Control

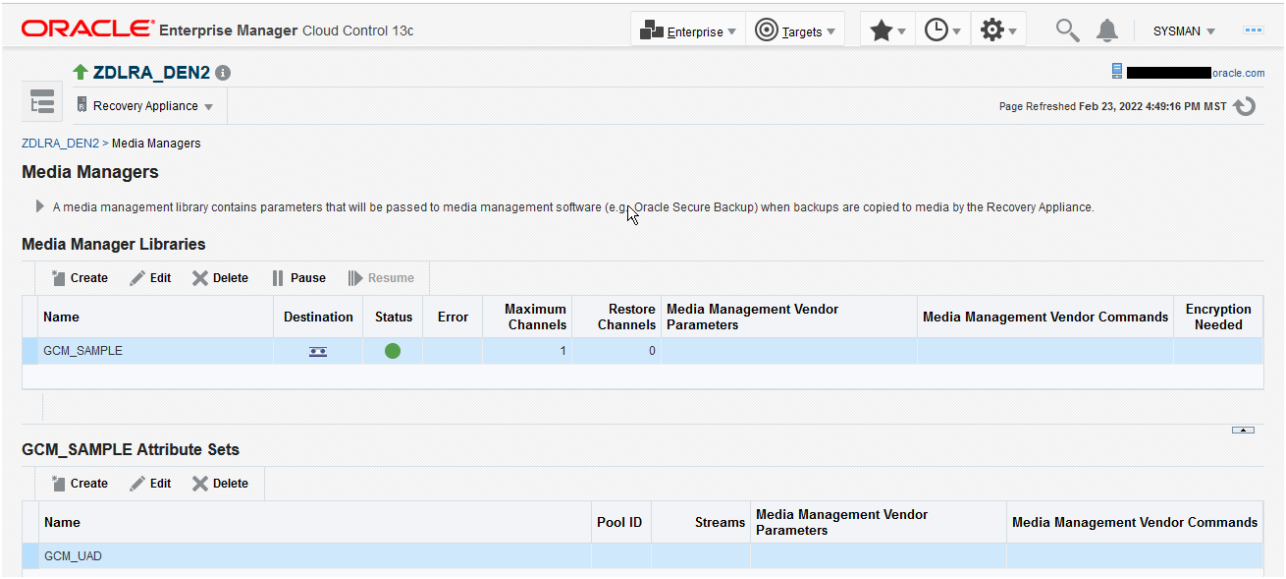
To create a media manager library with Cloud Control:

1. Complete the steps in "Accessing the Recovery Appliance Home Page".

   On the Recovery Appliance Home page, select **Media Managers** from the Recovery Appliance Menu. It displays the Media Managers screen with the default Oracle Secure Backup library and its corresponding attribute sets.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Figure 9-1 Media Managers Page



2. On the Media Managers page, click **Create** to configure a new media manager library.

   The Create Media Manager Library and Initial Attribute Set dialogue box appears.

3. In the **Media Manager Library** section, enter a name for this library.

4. In the **Maximum Channels** field, select the maximum number of media channels this media manager library can access.

5. Optionally, you can choose to enter **Advanced Parameters** for this media manager library.

   a. In the **Restore Channels** field, specify the number of media channels that you want to reserve for restore operations. If you do not enter any restore channel value, then the current restore operation uses the first free media channels that is available once all the backup operations are complete.

   b. In the **Media Management Vendor Parameters**, you can choose to add additional parameters to define your media manager library.

      For example, a media manager vendor parameter for Oracle Secure Backup contains the `SBT_LIBRARY` parameter by default, which specifies the path of the media manager library.

   If you are using a third party product as your media manager, create a new media library and use product-specific parameters for the specified media manager, especially the `SBT_LIBRARY` location parameter.

6. To add the initial attribute set for this library, complete the steps in the section "Creating an Attribute Set".

   If you do not enter any values for the attribute set, default values are applied.

7. Click **OK**.

> **See Also**
>
> "Recovery Appliance Components for Managing Tape Operations"

Copyright © 2014, 2026, Oracle and/or its affiliates.

Command_Line

To create a media manager library with DBMS_RA:

1. Start SQL*Plus or SQL Developer, and then log in to the metadata database as `RASYS` or as a named `db_user` with `user_type=admin`.
2. Run the `DBMS_RA.CREATE_SBT_LIBRARY` procedure.

```
BEGIN
   DBMS_RA.CREATE_SBT_LIBRARY(
     lib_name       => 'osbsbt',
     drives         => 12,
     restore_drives => 2,
     parms          => 'SBT_LIBRARY=libobk.so');
END;
```

In this example, the media management software is Oracle Secure Backup. The drives argument specifies the maximum number of tape drives that this SBT library can access. The `restore_drives` argument sets the number of tape drives that will be reserved for restore operations. The `parms` argument has the same purpose and format as the `PARMS` clause of an RMAN `ALLOCATE CHANNEL` command. It typically includes at least the `SBT_LIBRARY` parameter. In this case it designates the shared library for the Oracle Secure Backup media family.

> **See Also**
>
> "CREATE_SBT_LIBRARY" for descriptions of procedure arguments

9.2.2 Creating an Attribute Set

An SBT attribute set is referenced by an SBT job and is a collection of attributes that controls a tape backup operation. The attribute set specifies the SBT library to use for the copy operation. It also specifies SBT channel parameters and parameters to pass to the media management software library. These parameters are merged with the parameters specified in the SBT library object.

A SBT attribute set helps you to further customize and categorize your backups while copying them to tape. An attribute set is created for each tape drive associated with its media manager library. It helps classify backups while storing them on tape by specifying parameters such as the media pool number, streams, and media manager commands needed to perform the tape backup operation.

> **Note**
>
> If all SBT attribute sets share the same parameter value, then you can specify that parameter in the SBT library object instead of in each SBT attribute set.

Recovery Appliance comes with Oracle Secure Backup pre-configured as its media manager. Oracle Secure Backup components which include a media manager library and attribute sets for each of its tape drives, are also preconfigured. The preconfigured attribute sets are typically named `DRIVE_COUNT_1`, `DRIVE_COUNT_2`, and so on for the number of existing tape drives.

Copyright © 2014, 2026, Oracle and/or its affiliates.

This section describes how to create additional attribute sets for third-party media managers using either Cloud Control or DBMS_RA.

- Cloud_Control
- Command_Line

Cloud_Control

To create an SBT attribute set with Cloud Control:

You create the initial attribute set for a media manager library while creating the media manager library itself. If you leave the initial attribute set fields empty while configuring the media manager library, the default values are used.

You can also use the following steps to create additional attribute sets for a third-party media manager library.

1. Complete the steps in "Accessing Recovery Appliance in Cloud Control".

   From the Recovery Appliance menu, select **Media Managers**.
2. Under the Attribute Sets section, click **Create**.

   The Create Attribute Set box appears.
3. In the **Name** field, enter a name for this attribute set.
4. Optionally, in the **Pool ID** field, enter the media pool number that will be used to store backup copies.
5. Optionally, in the **Streams** field, specify the maximum number of streams that will be used to perform the tape backup operation.

   If you do not enter any value, then all available streams will be used.
6. Optionally, use the **Media Management Vendor Parameters** field to specify additional parameters to define your tape backup job. The Recovery Appliance uses a combination of these parameters and the media manager library parameters to complete the tape backup job.
7. Optionally, in the **Media Management Vendor Commands** field, enter vendor-specific commands for your media manager software to control your tape backup job.
8. Click **OK**.

Command_Line

To create an SBT attribute set with DBMS_RA:

1. With SQL*Plus or SQL Developer, connect to the Recovery Appliance database as an administrator of the Recovery Appliance.
2. Run the `DBMS_RA.CREATE_SBT_ATTRIBUTE_SET` procedure for each SBT attribute set that you want to create.

```
BEGIN
  DBMS_RA.CREATE_SBT_ATTRIBUTE_SET(
    lib_name             => 'osbsbt',
    attribute_set_name   => 'wholedb',
    streams              => 10,
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
    parms               => 'ENV=(OB_MEDIA_FAMILY=wholedb_mf)');
END;
```

Again in this example, the media management software is Oracle Secure Backup (OSB). The streams argument sets the maximum number of concurrent streams that can be used for automated backups. The `parms` argument has the same purpose and format as the `PARMS` clause of an RMAN `ALLOCATE CHANNEL` command. In this case it designates the `wholedb_mf` Oracle Secure Backup media family as the destination of the copy operation.

See Also

- "CREATE_SBT_ATTRIBUTE_SET" for descriptions of procedure arguments
- "Backup Retention on Tape" for a discussion of the `sbt_retention_policy` argument

9.3 Managing Tape Backup Job Components

After you have created your tape backup job components, you may need to modify their properties periodically based on job requirements or delete them when no longer required.

This section contains the following topics:

- **Enterprise Manager Cloud Control**
  - Managing a Media Manager Library Using Cloud Control
  - Managing an Attribute Set Using Cloud Control
- **PL/SQL Package DBMS_RA**
  - Managing an SBT Library Using DBMS_RA
  - Managing an Attribute Set Using DBMS_RA

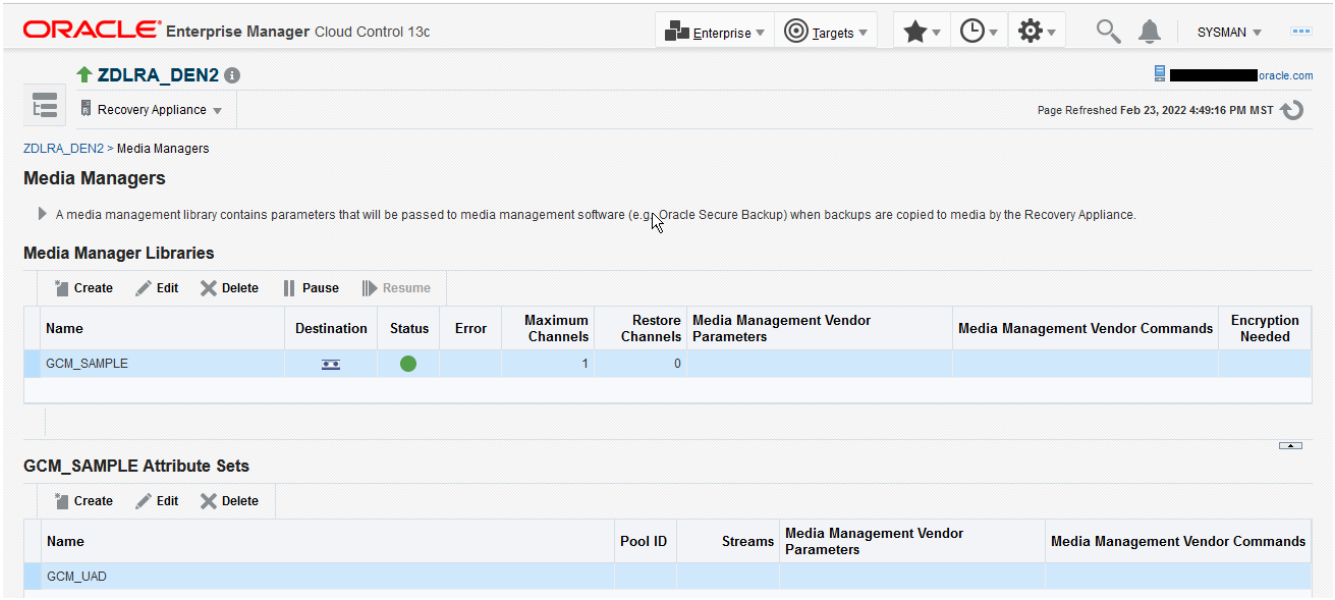
9.3.1 Managing a Media Manager Library Using Cloud Control

You can edit the existing properties of a media manager library to update parameters based on your tape backup job requirements.

You can delete an existing media manager library after all tape backup operations associated with this media manager are complete and you no longer require the parameters specified as a part of this library.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Figure 9-2 Media Managers Page



- Edit
- Delete

Edit

To edit a Media Manager library:

1. Complete the steps in "Accessing the Recovery Appliance Home Page".

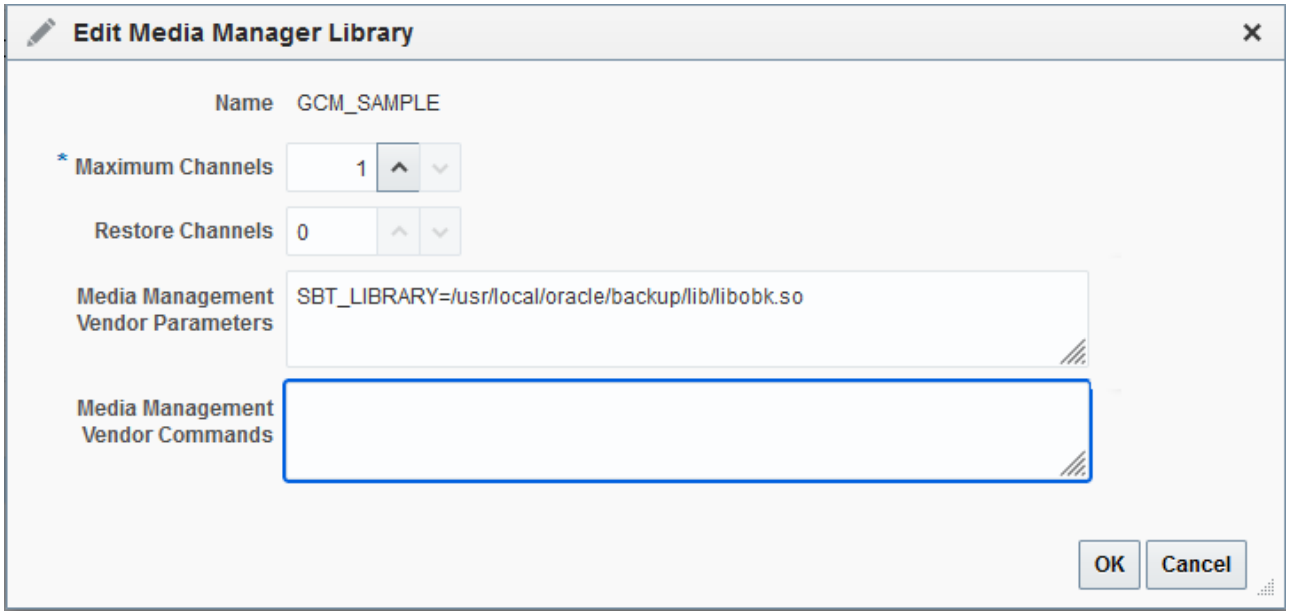
   On the Recovery Appliance Home page, select **Media Managers** from the Recovery Appliance menu.

2. On the Media Managers page, select the Media Manager library that you want to edit and click **Edit**.

   The Edit Media Manager Library screen appears with the existing library parameters.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Figure 9-3 Edit Media Manager Library Screen



3. Change the number of **Maximum Channels**.
4. Click **OK**.

Delete

To delete a Media Manager library:

1. Complete the steps in "Accessing Recovery Appliance in Cloud Control".
2. From the Recovery Appliance Menu, select **Media Managers**.
3. From the list of existing media manager libraries, select the Media Manager Library you want to delete.
4. Click **Delete**.

   A message asks you confirm the deletion of this library.
5. Click **Yes**.

9.3.2 Managing an Attribute Set Using Cloud Control

This section contains information on how to edit or delete an attribute set using Cloud Control.

- Edit
- Delete

Copyright © 2014, 2026, Oracle and/or its affiliates.

Edit

To edit an attribute set:

You can edit the existing properties of a media manager attribute set to modify your tape backup job settings at a job-specific level.

1. Complete the steps "Accessing Recovery Appliance in Cloud Control".
2. From the Recovery Appliance Menu, select **Media Managers**.
3. From the list of attribute sets, select one attribute set that you need to edit.
4. Make the required changes to the **Pool ID**, **Media Management Vendor Parameters**, and **Media Management Vendor Command** values.
5. Click **OK**.

Delete

To delete an attribute set:

You can delete an existing attribute set after all associated tape backup jobs are complete and you no longer require the job parameters specified in the attribute set.

1. Complete the steps in "Accessing Recovery Appliance in Cloud Control".
2. From the Recovery Appliance Menu, select **Media Managers**.
3. In the Attribute Sets section, select the attribute list that you want to delete.
4. Click **Delete**.

   A message asks you to confirm deletion of this attribute set.
5. Click **Yes**.

9.3.3 Managing an SBT Library Using DBMS_RA

This section contains information on how to use `DBMS_RA` to edit or delete an SBT library.

- Edit
- Delete

Edit

To edit an SBT library:

Delete

To delete an SBT library:

You can delete an SBT library by calling the given procedure in the `DBMS_RA` PL/SQL package.

1. Using SQL*Plus or SQL Developer, connect to the Recovery Appliance metadata database as the Recovery Appliance administrator.

Copyright © 2014, 2026, Oracle and/or its affiliates.

2. Run the DELETE_SBT_LIBRARY procedure, providing the name of the SBT object to delete.

```
BEGIN
  DBMS_RA.DELETE_SBT_LIBRARY(
    lib_name => 'OSBSBT');
END;
```

See Also

DBMS_RA Package Reference

9.3.4 Managing an Attribute Set Using DBMS_RA

This section contains information on how to edit or delete an attribute set using DBMS_RA.

- Edit
- Delete

Edit

To edit an attribute set:

You can modify one or more attributes of an SBT attribute set by calling the given procedure in the DBMS_RA PL/SQL package.

1. Using SQL*Plus or SQL Developer, connect to the Recovery Appliance database as the Recovery Appliance administrator.
2. Run the UPDATE_SBT_ATTRIBUTE_SET procedure, providing the name of the SBT attribute set to modify and the new values for its attributes.

   Attributes omitted from the procedure call are left unchanged.

Delete

To delete an attribute set:

You can delete an SBT attribute set by calling the given procedure in the DBMS_RA PL/SQL package.

1. Using SQL*Plus or SQL Developer, connect to the Recovery Appliance database as the Recovery Appliance administrator.
2. Run the DELETE_SBT_ATTRIBUTE_SET procedure, providing the name of the SBT object to delete.

Copyright © 2014, 2026, Oracle and/or its affiliates.

See Also

DBMS_RA Package Reference

9.4 Creating a Tape Backup Job

This section describes how to create tape backup jobs for a selected database or databases associated with a protection policy. The tape backup job templates define the properties for backups that need to be stored on tape. Media manager libraries and their attribute sets manage these job settings.

Recovery Appliance gives you the option to copy backups from multiple protected databases to tape. To perform this task, specify the protection policy that contains the protected databases for which you want to copy backups to tape. Alternatively, you can also copy backups to tape for a single specified database.

You can create a tape backup job using Cloud Control or command-line options.

- Cloud_Control
- Command_Line

Cloud_Control

To create a tape backup job with Cloud Control:

1. Complete the steps in "Accessing the Recovery Appliance Home Page"
2. On the Recovery Appliance Home page, from the Recovery Appliance menu, select **Copy-To-Media Job Templates**.

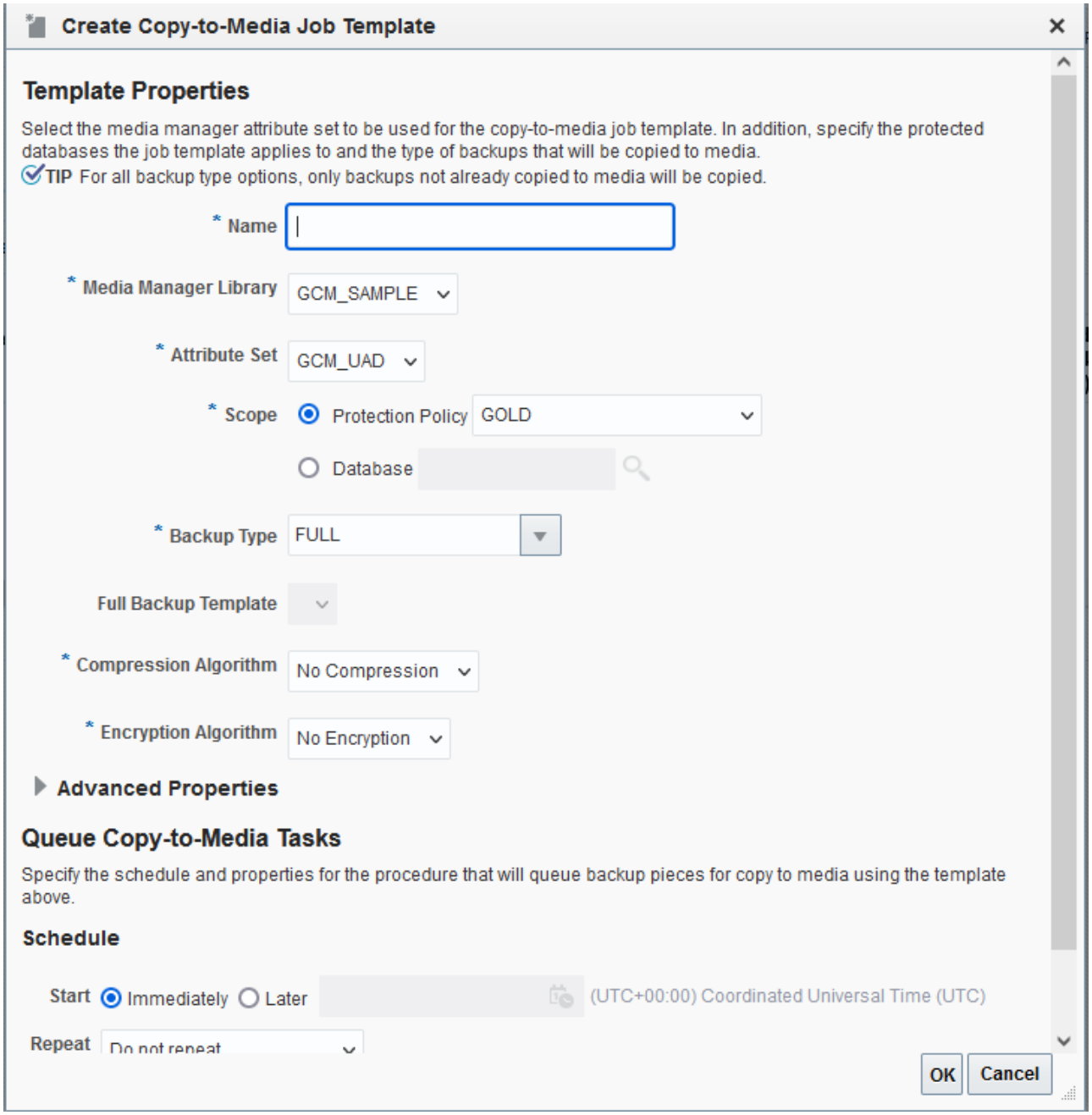
   The Copy-to-Media Job Templates page appears.
3. Click **Create** to create a new media backup job.

   The Create Copy-to-Media Job Template page appears as shown in #unique_284/unique_284_Connect_42_CHDJJCEB.

   This page displays the job properties and schedule settings.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Figure 9-4 Recovery Appliance Create Copy-to-Media Job Template Page



4. In the **Name** field, enter a name for the job.
5. Optionally, from the **Media Manager Library** drop-down list, select a media manager library that will manage this job.
6. From the **Attribute Set** drop-down list, select an attribute set that you want to use for this job. This attribute set will define the settings for your tape backup job.
7. In the **Scope** field, add one of the following:
   - Select the protection policy from the **Protection Policy** drop-down list that includes all the databases for which you want to copy the backups to tape.
   - Search for and select a single database for which you want to copy the backups to tape.

Copyright © 2014, 2026, Oracle and/or its affiliates.

8. In the **Backup Type** field, select the type of backup to be copied. Options include: Full backup, Incremental Backup, and Archived Log.
9. (Optional) In the **Advanced Properties** area, use the **Priority** drop-down to specify the priority for this job. The default job priority is Medium.
10. In the **Advanced Properties** area, use the **Number of copies** drop-down to specify the number of copies you need for the backup being copied to media. You can choose a maximum of 4 copies and a minimum of 1 copy.

> **Note**
>
> You cannot obtain duplicate copies of a backup after it has been copied to tape.

11. In the **Advanced Properties** area, use the **Runtime Window** field to specify the amount of time in minutes, hours or days allowed for this job to complete. Jobs that do not start within the specified window will be completed in the next available window slot.
12. In the **From Tag** field, specify a tag name to only copy backups associated with a certain tag to tape.
13. Under the **Schedule** section, specify whether you want this job to run immediately or at a later specified time.

> **See Also**
>
> "Scheduling a Tape Backup Job"

14. In the **Properties** section, you can specify optional properties that are applied to the backup pieces being copied to media. These include **Format**, **Autobackup Prefix**, and **Tag**.
15. Click **OK**.

    Cloud Control displays a message notifying that your job request has been submitted successfully.

    You can click the job name in the message to check the queued backup images for this job.

Command_Line

To create a tape backup job with DBMS_RA:

Each SBT job defines the backups to be copied to tape and the media pool to which to copy them. After you create SBT jobs, you must schedule their execution.

1. Log in to SQL*Plus or SQL Developer with an `admin db_user` user.
2. Run the CREATE_SBT_JOB_TEMPLATE procedure for each SBT job that you want to create.

Copyright © 2014, 2026, Oracle and/or its affiliates.

> **Note**
>
> `CREATE_SBT_JOB_TEMPLATE` is an overloaded procedure. With one procedure signature, you specify the `db_unique_name` of a protected database for which to consider backups for copying. With the other procedure signature, shown below, you specify a protection policy name. In this case, backups for all protected databases assigned to the protection policy are considered for copying.

3. If a protection policy name is specified with the `protection_policy_name` parameter, then when the SBT job runs, backups for all databases assigned to the protection policy are considered for copying to tape. If a `db_unique_name` is specified, then only backups for that database are considered for copying.
4. Using the `attribute_set_name` parameter, specify the name of an SBT attribute set, which is a collection of attributes that control tape backup operations. The attribute set specifies the SBT library to use for the copy operation. It also specifies SBT channel parameters and parameters to pass to the media management software library. These parameters are merged with the parameters specified in the SBT library object.
5. Using the `backup_type` parameter, add the types of backups to copy, expressed as a comma-delimited list of the following types: `ALL`, `INCR`, `ARCH`, or `FULL`. For example, if `'INCR,ARCH'` is specified, then all incremental (level 1) backups and archived log files that have not yet been copied to the named media manager are included.
6. With the `priority` parameter, enter a priority level for this job. When many SBT jobs are scheduled to run simultaneously, the job priority determines the job that runs first. Job priority is needed when there are not enough tape drives to service all of the jobs that are scheduled to run simultaneously. Job priority is expressed as one of the following predefined values:
   - `1000 (SBT_PRORITY_LOW)`
   - `100 (SBT_PRIORITY_MEDIUM)`
   - `10 (SBT_PRIORITY_HIGH)`
   - `1 (SBT_PRIORITY_CRITICAL)`

   `0` is the highest possible priority. Lower priority values take precedence over higher values. The default priority is `100 (SBT_PRIORITY_MEDIUM)`.

Example of Creating a Tape Backup Job

- Cloud_Control
- Command_Line

Cloud_Control

To create a tape backup job with Cloud Control:

This example uses a combination of tape backup jobs to manage copying all your backups to tape and ensure that they are up-to-date. The combination of tape backup jobs stores all your backups on tape systematically and reduces chances of loss of information.

Copyright © 2014, 2026, Oracle and/or its affiliates.

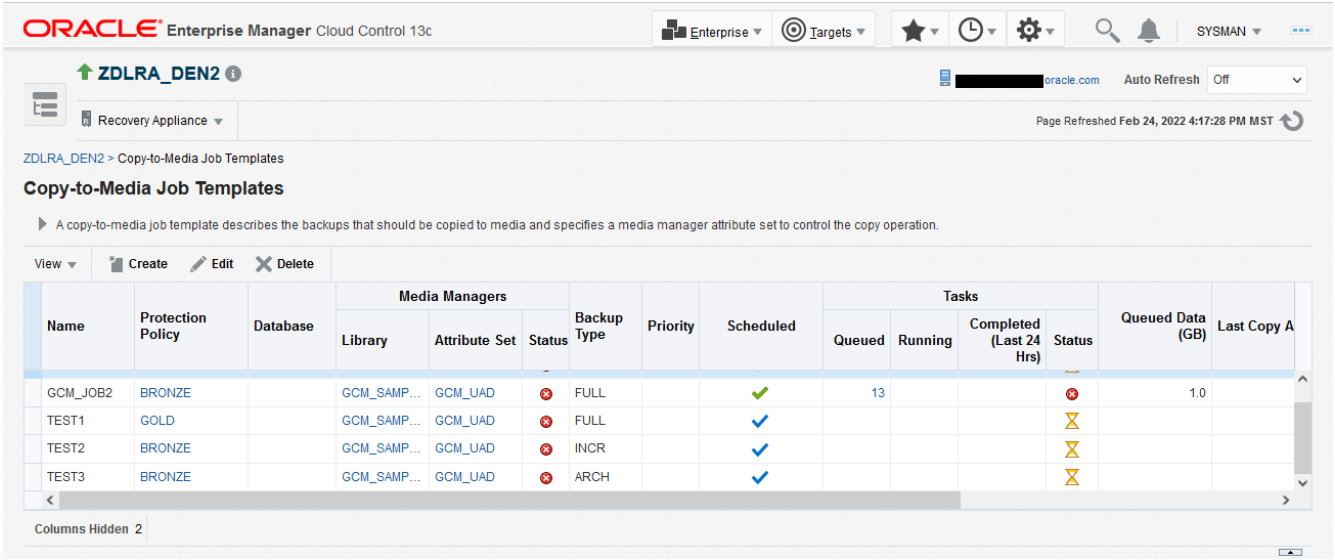
1. Create your first tape backup job and name this job `Test1`.
2. Use the default media manager library (example `GCM_SAMPLE`) and the attribute set (example `GCM_UAD`) for this job.

   Edit the default media manager library values and default attribute set values, as per "Managing Tape Backup Job Components".
3. Select the scope of this job as the Protection Policy `GOLD`.
4. Select **Full** as the Backup Type for this job.

   This selection implies that all your full database backups for databases included in `GOLD` will be copied to tape.
5. Schedule `Test1` to run on Sunday at 11:11 am and set it to repeat every 1 week.
6. Similarly, create a second tape backup job and name this job `Test2`.
7. Let this job use the same media manager library, protection policy and attribute set as `Test1`.
8. Select **INCR** (incremental) from the **Backup Type** drop-down list for this job.
9. Schedule `Test2` to run from Monday to Saturday at 12:30 pm and set it to repeat daily.
10. Using the same steps as above, create a third tape backup job and name it `Test3`.
11. Select **ARCH** (Archived Logs) as the **Backup Type** for this job.
12. Schedule `Test3` to run at 6 hour intervals.

#unique_285/unique_285_Connect_42_BABJEDBC displays the Recovery Appliance Copy-to-Tape Templates screen after these jobs have been successfully submitted.

Figure 9-5 Tape Backup Jobs Example



In this scenario, the Recovery Appliance copies all full backups to tape once a week. Afterward, the Recovery Appliance copies all incremental backups with the latest changes daily to maintain the updated backup copies on tape. Similarly, the system copies all archived redo log files every 6 hours to ensure efficient copy and storage of backups on tape.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Command_Line

To create a tape backup job with DBMS_RA:

This example illustrates how to create an SBT job using the `CREATE_SBT_JOB_TEMPLATE` procedure.

```
BEGIN
  DBMS_RA.CREATE_SBT_JOB_TEMPLATE (
    template_name           => 'oltp_arch_lastfull',
    protection_policy_name  => 'oltp',
    attribute_set_name      => 'wholedb',
    backup_type             => 'FULL,ARCH',
    priority                => DBMS_RA.SBT_PRIORITY_HIGH,
    window                  => INTERVAL '4' HOUR);
END;
```

In this example, the SBT job selects all archived log files and the last full backup for every protected database assigned to the `oltp` protection policy. The last full backup could be either the most recent level 0 backup received, or a virtual full backup based on the most recent level 1 backup received, whichever is later.

The SBT job references the `wholedb` SBT attribute set, which specifies the SBT library to use for the copy operation, and specifies SBT channel parameters and parameters to pass to the media management software library.

The `backup_type` parameter copies all archived log backups not yet copied to SBT and the most recent full backup, if it has not already been copied to tape. For example, the `backup_type` of "full, arch" selects all archived log backups, and the most recent full backup. These are the backups that will be copied to SBT when this job is run, if they have not already been copied.

The four-hour window specifies the length of time that copy tasks generated by this job are eligible to be started. When the window expires, any SBT copy tasks that were generated by this job but not yet started will be suspended until the next time this SBT job is scheduled. Copy tasks that are already running when the window expires are allowed to complete.

Not shown are the optional `copies` and `from_tag` arguments.

9.5 Managing a Tape Backup Job

After you have created a tape backup job, you may need to modify some of its properties based on job requirements or delete it when no longer required.

> **See Also**
>
> "Creating a Tape Backup Job" for more information on how to create a tape backup job

- Cloud_Control
- Command_Line

Copyright © 2014, 2026, Oracle and/or its affiliates.

Cloud_Control

To manage a tape backup job with Cloud Control:

Edit Tape Backup Job

1. Complete the steps in "Creating a Tape Backup Job".
2. Select the tape backup job that you want to edit.
3. Click **Edit** to change the existing job properties and schedule settings.
4. Make the required changes to the existing **Media Manager Library**, **Attribute Set**, **Priority**, **Copies**, **Runtime Window**, and **From Tag** values, to edit job properties.
5. Make changes the existing schedule options, to edit the job schedule settings.
6. Click **OK**.

Delete Tape Backup Job

1. Complete the steps in "Accessing the Recovery Appliance Home Page".
2. On the Recovery Appliance Home page, from the Recovery Appliance menu, select **Copy-to-Tape Job Templates**.

   The Copy-to-Tape Job Template page appears.
3. From the list of jobs, select the job that you want to delete.
4. Click **Delete**.

   A confirmation message appears asking whether you want to continue with deleting this job. Click **Yes**.

Command_Line

To manage a tape backup job with DBMS_RA:

Edit Tape Backup Job

You can modify one or more attributes of an SBT job by calling the given procedure in the `DBMS_RA` PL/SQL package.

1. Using SQL*Plus or SQL Developer, connect to the Recovery Appliance database as the Recovery Appliance administrator.
2. Run the UPDATE_SBT_JOB_TEMPLATE procedure, providing the name of the SBT job to modify and the new values for its attributes.

   This example modifies the priority of the SBT job named `oltp_arch_lastfull`. The values of other arguments present in CREATE_SBT_JOB_TEMPLATE and omitted in this call remain unchanged.

```
BEGIN
  DBMS_RA.UPDATE_SBT_JOB_TEMPLATE(
    template_name => 'oltp_arch_lastfull',
    priority      => DBMS_RA.SBT_PRIORITY_HIGH);
END;
```

Delete Tape Backup Job

You can delete a SBT job by calling the given `DBMS_RA` PL/SQL package procedure.

Copyright © 2014, 2026, Oracle and/or its affiliates.

1. Using SQL*Plus or SQL Developer, connect to the Recovery Appliance metadata database as the Recovery Appliance administrator.
2. Run the DELETE_SBT_JOB_TEMPLATE procedure, providing the name of the SBT job to delete.

> **See Also**
>
> DBMS_RA Package Reference

9.6 Scheduling a Tape Backup Job

This section describes the steps required to set the schedule for a tape backup job.

- Cloud_Control
- Command_Line

Cloud_Control

To schedule a tape backup job with Cloud Control:

1. Complete the steps in "Accessing Recovery Appliance in Cloud Control".
2. In the **Schedule** section, select one of the following:
   - **Immediately**: Runs the tape backup job right away.
   - **Later**: Runs the tape backup job at a future, specified time. Enter the date and time when you want the job to run in the `mm/dd/yyyy hh:mm:ss` format. If you want this job to repeat, then select the frequency from the **Repeat** drop-down list and enter the applicable values.
3. Click **OK** to save the job.

   To edit the schedule settings, select the required tape backup job, click **Edit**, and make the necessary changes.

Command_Line

To schedule a tape backup job with DBMS_RA:

After creating an SBT job, you must schedule its execution. Typically, you schedule the job to run at regular intervals using a scheduling utility such as Oracle Scheduler.

When an SBT job runs, Recovery Appliance examines all backups of the specified type for the specified databases, and selects the ones that have not yet been copied to tape. Recovery Appliance then generates and queues tasks that copy those backups to tape. One task of type `BACKUP_SBT` is added to the Recovery Appliance task queue for each backup piece to copy.

To schedule a tape backup job with Oracle Scheduler, you create a job that will invoke `DBMS_RA.QUEUE_SBT_BACKUP_TASK`. This procedure takes a single argument, the name of the SBT job.

Copyright © 2014, 2026, Oracle and/or its affiliates.

1. With SQL*Plus or SQL Developer, connect to the Recovery Appliance database as the Recovery Appliance administrator.
2. Run `DBMS_SCHEDULER.CREATE_JOB`.

   For example, run the following PL/SQL anonymous block:

```
BEGIN
  DBMS_SCHEDULER.CREATE_JOB(
   job_name   => 'sbtjob1',
   job_type        => 'PLSQL_BLOCK',
   job_action      => 'dbms_ra.queue_sbt_backup_task(''oltp_arch_lastfull'');',
   start_date      => SYSTIMESTAMP,
   enabled         => TRUE,
   auto_drop       => TRUE,
   repeat_interval => 'freq=WEEKLY; BYDAY=MON,WED,FRI; BYHOUR=23');
END;
```

See Also

- "Creating a Tape Backup Job" to learn how to create an SBT job
- "QUEUE_SBT_BACKUP_TASK"
- See *Oracle AI Database Administrator's Guide* and *Oracle AI Database PL/SQL Packages and Types Reference* for information about Oracle Scheduler

9.7 Pausing and Resuming Tape Backup Operations

This section describes the steps to pause an ongoing tape backup job or media manager operations and how to resume them later.

- Cloud_Control
- Command_Line

Cloud_Control

To pause and resume tape backup operations with Cloud Control:

You pause ongoing media manager library operations in a situation where you want to put the current media manager library on hold and resume it after a certain period.

1. Complete the steps in "Accessing Recovery Appliance in Cloud Control".
2. From the Recovery Appliance Menu, select **Media Managers**.
3. From the list of media manager libraries, select the library for which you want to pause all ongoing jobs.
4. Click **Pause**.

You can resume operations of any paused media manager library for the library and its associated job operations to continue.

Copyright © 2014, 2026, Oracle and/or its affiliates.

1. Complete the steps in "Accessing Recovery Appliance in Cloud Control".
2. From the Recovery Appliance Menu, select **Media Managers**.
3. Select the paused media manager library for which you want to resume job operations.
4. Click **Resume**.

Command_Line

To pause and resume tape backup operations with DBMS_RA:

Pause Tape Backup

1. With SQL*Plus or SQL Developer, connect to the Recovery Appliance database as the Recovery Appliance administrator.
2. Run the `DBMS_RA.PAUSE_SBT_LIBRARY` procedure.

   For example, run the following PL/SQL anonymous block:

   ```
   BEGIN
     DBMS_RA.PAUSE_SBT_LIBRARY(
       lib_name => 'osbsbt');
   END;
   ```

   The library status in the view `RA_SBT_LIBRARY` changes to `PAUSE`.

Resume Tape Backup

1. With SQL*Plus or SQL Developer, connect to the Recovery Appliance metadata database as the Recovery Appliance administrator.
2. Run the `DBMS_RA.RESUME_SBT_LIBRARY` procedure.

   For example, run the following PL/SQL anonymous block:

   ```
   BEGIN
     DBMS_RA.RESUME_SBT_LIBRARY(
       lib_name => 'osbsbt');
   END;
   ```

The library status in the view `RA_SBT_LIBRARY` changes back to `READY`.

See Also

- "PAUSE_SBT_LIBRARY"
- "RESUME_SBT_LIBRARY"
- "RA_SBT_LIBRARY"

9.8 Viewing the Status of Tape Backup Operations

This section describes the steps to check the status of media manager library and tape backup jobs to monitor the progress of their operations.

This section contains the following topics:

Copyright © 2014, 2026, Oracle and/or its affiliates.

- Viewing the Status of Tape Backup Operations Using Cloud Control
- Viewing the Status of Tape Backup Operations Using DBMS_RA

9.8.1 Viewing the Status of Tape Backup Operations Using Cloud Control

This section contains the following topics:

- Viewing the Media Manager Library Status
- Viewing the Tape Backup Job Status

9.8.1.1 Viewing the Media Manager Library Status

You can check for any errors in the media management software that may affect the ongoing tape backup operations by viewing the current status of a media manager library.

To check the status of a media manager library:

1. Complete the steps "Accessing Recovery Appliance in Cloud Control".
2. From the Recovery Appliance Menu, select **Media Managers.**

   The Media Managers page lists the status of all existing media manager libraries in the corresponding column.

9.8.1.2 Viewing the Tape Backup Job Status

You can check for any error in the ongoing tape backup operation by viewing the current status of the job.

To view the status of a tape backup job:

1. Complete the steps "Accessing Recovery Appliance in Cloud Control".
2. From the Recovery Appliance Menu, select **Copy-to-Tape Job Templates**.

   The Copy-to-Tape Job Templates Page displays the status of each tape backup job in its corresponding column.

9.8.2 Viewing the Status of Tape Backup Operations Using DBMS_RA

This section contains the following topics:

- Checking the SBT Library Status
- Checking the Tape Backup Job Status
- Reviewing SBT Job Runs Using DBMS_RA
- Checking the Status of Oracle Scheduler Jobs

9.8.2.1 Checking the SBT Library Status

You can determine if an error condition exists in the media management software by querying the `RA_SBT_LIBRARY` view. You can also determine the `PAUSE` state of an SBT library.

To view the status of an SBT library:

1. With SQL*Plus or SQL Developer, connect to the Recovery Appliance metadata database as the Recovery Appliance administrator.

Copyright © 2014, 2026, Oracle and/or its affiliates.

2. Run the following query:

```
SELECT LIB_NAME, LAST_ERROR_TEXT, STATUS
FROM   RA_SBT_LIBRARY;

LIB_NAME            LAST_ERROR_TEXT                                  STATUS
------------------- ------------------------------------------------ ------
OSBSBT                                                               READY
```

Table 9-3 lists the possible values for STATUS.

Table 9-3 Values for the STATUS Column of RA_SBT_LIBRARY

Value	Meaning
READY	The SBT library was properly created and is ready to process tape I/O.
PAUSE	The SBT library was paused with PAUSE_SBT_LIBRARY.
ERROR	An error condition exists in the media management software. Tape backup operations cannot continue until you clear the error and call RESUME_SBT_LIBRARY. The last_error_text column describes the most recent error returned by the media management library. The error text also appears in the RA_SBT_TASK rows for the affected background SBT tasks.

See Also

"RA_SBT_LIBRARY"

9.8.2.2 Checking the Tape Backup Job Status

When an SBT job runs it generates and queues a task for each backup piece to be copied. The view RA_SBT_TASK lists these tasks and their completion states.

Note

Completed tasks are removed from this view after 30 days.

To check the status of SBT jobs:

1. With SQL*Plus or SQL Developer, connect to the Recovery Appliance metadata database as the Recovery Appliance administrator.
2. Run the following query (sample output shown):

```
SELECT TASK_ID, STATE, DB_UNIQUE_NAME, ERROR_TEXT, BS_KEY, PIECE#
FROM   RA_SBT_TASK
WHERE  SBT_TEMPLATE_NAME = 'SBTJOB1'
ORDER BY DB_UNIQUE_NAME, BS_KEY, PIECE#;

TASK_ID STATE      DB_UNIQUE_NAME ERROR_TEXT           BS_KEY PIECE#
------- ---------- -------------- -------------------- ------ ------
    253 COMPLETED  OLTP1                                    89      1
    254 COMPLETED  OLTP1                                    95      1
    255 COMPLETED  OLTP1                                    99      1
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
       256 COMPLETED  OLTP1                                    117      1
       257 COMPLETED  OLTP1                                    141      1
```

A task failed if its `STATE` is `KILLED` or `ABORTED`.

9.8.2.3 Reviewing SBT Job Runs Using DBMS_RA

You can determine whether SBT jobs run according to your defined schedule by querying the `RA_SBT_JOB.LAST_SCHEDULE_TIME` column. This column indicates the last time that the SBT job was scheduled to run.

To review completed SBT jobs:

1. With SQL*Plus or SQL Developer, connect to the Recovery Appliance metadata database as the Recovery Appliance administrator.
2. Run the following query (sample output shown):

```
SELECT TEMPLATE_NAME, BACKUP_TYPE, LAST_SCHEDULE_TIME
FROM   RA_SBT_JOB;

TEMPLATE_NAME          BACKUP_TYPE          LAST_SCHEDULE_TIME
-----------------  -------------------  ---------------------------
OLTP_ARCH_LASTFULL  ARCH, FULL          27-AUG-12 10.53.49 AM -08:00
OLTP_INCR           INCR                26-AUG-12 10.16.16 AM -08:00
```

> **See Also**
>
> "RA_SBT_LIBRARY"

9.8.2.4 Checking the Status of Oracle Scheduler Jobs

To view the status of an Oracle Scheduler job created to run an SBT job:

- Query the following views:
  - `USER_SCHEDULER_JOBS`
  - `USER_SCHEDULER_JOB_LOG`

> **See Also**
>
> *Oracle Database Administrator's Guide* for details about data dictionary views for Oracle Scheduler

Copyright © 2014, 2026, Oracle and/or its affiliates.

10

Converting Tape Configurations to Cloud

This chapter explains how to convert existing Oracle Secure Backup tape configurations to equivalent cloud configurations using the `racli convert tape_to_cloud` command.

This chapter contains these topics:

- About Tape to Cloud Conversion
- Requirements for Converting Tape Configurations to Cloud
- Converting a Tape Configuration to Cloud

10.1 About Tape to Cloud Conversion

The `racli convert tape_to_cloud` command provides an automated method to convert existing Oracle Secure Backup (OSB) tape configurations into equivalent cloud-based configurations. The `racli convert tape_to_cloud` command transforms tape-based attribute sets, backup templates, and associated scheduling constructs into corresponding cloud configurations, ensuring a consistent transition of backup operations.

The conversion preserves existing naming conventions, retention settings, and protection policies so that backup behavior remains unchanged after the transition. By maintaining configuration integrity and consistency, the process minimizes operational impact and eliminates the need for manual reconfiguration.

The conversion operates entirely at the configuration level and does not move existing backup data. Original tape configurations are retained on the appliance so that you can restore historical tape backups, if needed.

Copyright © 2014, 2026, Oracle and/or its affiliates.

> **Note**
>
> Oracle Secure Backup (OSB) 19.1 desupport was announced on May 1, 2026 with premier support end date on September 30, 2027. See My Oracle Support note PNEWS3035 for more information. Please refer to Lifetime Support Policy: Oracle Technology Products for updated support timeframes.
>
> Oracle recommends that database backups to cloud take advantage of Oracle Database Zero Data Loss Cloud Protect, which uses Oracle Zero Data Loss Autonomous Recovery Service in OCI. Cloud Protect offers an efficient incremental forever backup strategy, real-time transaction protection, logically air-gapped immutable backups, and fast, point-in-time recovery.
>
> For customers who require database backups to Amazon S3, Oracle Database Cloud Backup for Amazon S3 may be used. This offering is similar to Oracle Secure Backup Cloud Module in that it is fully integrated with Recovery Manager (RMAN) and enables you to back up your Oracle database to Amazon S3. You can exchange your Oracle Secure Backup license for an Oracle Database Cloud Backup for Amazon S3 license on a 1:1 basis. See Oracle Database Cloud Backup for Amazon S3 for more information.
>
> As with Oracle Secure Backup Cloud Module, Oracle Database Cloud Backup for Amazon S3 can be used free of charge on Oracle Zero Data Loss Recovery Appliance (ZDLRA) for the sole purpose of backing up data to and restoring data from S3-compatible storage devices that are in the same campus as ZDLRA.

10.2 Requirements for Converting Tape Configurations to Cloud

Verify these prerequisites before you run the `racli convert tape_to_cloud` command.

Table 10-1 Prerequisites for Tape to Cloud Conversion

Requirement	Description
Existing source SBT library for Oracle Secure Backup (OSB)	Ensure that the source SBT library for OSB exists and is accessible (for example, `ROBOT0`, `ROBOT1`), and the tape templates, configured for the SBT library, is in an active state.
Cloud key	Use the `racli add cloud_key` command to create a cloud key required for OCI access.
Cloud user	Create a cloud user using the `racli add cloud_user` command. Associate the cloud user with the cloud key.
Cloud storage location	Create an OCI Object Storage bucket. Add the bucket to Recovery Appliance using the `racli add cloud_location` command.

10.3 Converting a Tape Configuration to Cloud

Use these steps to configure the cloud resources, run the conversion, and verify that an existing tape configuration has been successfully converted to cloud.

Copyright © 2014, 2026, Oracle and/or its affiliates.

1. Create a cloud key.

   ```
   racli add cloud_key --key_name=<KEY_NAME>
   ```

   .

2. Register the key fingerprint.

   ```
   racli alter cloud_key --key_name=<KEY_NAME> --fingerprint=<FINGERPRINT>
   ```

3. Create a cloud user.

   ```
   racli add cloud_user --user_name=CLOUD_USER_NAME --key_name=CLOUD_KEY_NAME
   --user_ocid=UOCID --tenancy_ocid=TOCID [--compartment_ocid=COCID]
   ```

4. Run this command to create a new cloud storage location, such as an OCI Object Storage bucket.

   ```
   racli add cloud_location --cloud_user=CLOUD_USER_NAME
   --host=HOST_URL --bucket=OCI_BUCKET_NAME
   ```

5. Run the `racli list cloud_location` command to list the cloud location in Recovery Appliance. In this sample output, the cloud location name is `OCI_LOC1`.

   ```
   [root@ra01 ~]# racli list cloud_location --all
     Wed May  6 21:05:00 2026: Using log file /opt/
   oracle.RecoveryAppliance/log/racli.log
     Wed May  6 21:05:00 2026: Start: List Cloud Location
     Cloud Location OCI_LOC1
             Location Name: OCI_LOC1
                Immutable: FALSE
                  Archive: NO
      Archive After Backup: YES
                   Bucket: my_bucket
              Guaranteed: YES
                    Host: [nnnnnn01.oracle.com]
             Location ID: 123
                Streams: 8
                 User ID: OCI_USER
             SBT Library: OCI_LOC1
       Attribute Set Name: Default
             Backup Stream: 8

     Wed May  6 21:05:01 2026: End: List Cloud Location
   ```

   Run the `racli convert tape to cloud` command. Specify the source SBT library and the cloud location name.

   ```
   racli convert tape_to_cloud --tape_library=SBT_LIBRARY
   --cloud_location=CLOUD_LOCATION
   ```


Copyright © 2014, 2026, Oracle and/or its affiliates.

In this example, `ROBOT0` is an existing source SBT library and `OCI_LOC1` is the cloud location.

```
racli convert tape_to_cloud --tape_library=ROBOT0 --cloud_location=OCI_LOC1
```

6. With SQL*Plus or SQL Developer, connect to the Recovery Appliance as a named user with `RASYS` privileges.

   ```
   $ sqlplus rasys/<password>
   ```

7. Query the view `RA_SBT_JOB` to list the defined SBT templates.

   ```
   SELECT TEMPLATE_NAME FROM RA_SBT_JOB;
   ```

8. Run the archive-to-cloud job to verify the copy to cloud operation.

   ```
   SQL> exec dbms_ra.queue_sbt_backup_task('<COPY_TO_CLOUD_TEMPLATE_NAME>');

   PL/SQL procedure successfully completed.
   ```

9. Verify that cloud-based templates have been created and that backup jobs reference the new cloud templates.

> **Note**
>
> If an existing tape template name contains the word *TAPE*, it is replaced with *CLOUD* in the converted template name. For example, the template *TAPE_TPL_01* is converted and renamed to *CLOUD_TPL_01*. The original *TAPE_TPL_01* template remains unchanged.

> **Note**
>
> For detailed information about the RACLI commands, see the Recovery Appliance Command Line Interface Reference.

Copyright © 2014, 2026, Oracle and/or its affiliates.

11

Archiving Backups to Cloud

This procedure for archive-to-cloud builds on the techniques used for copy-to-tape. The difference is that it sends backups to cloud repositories for longer term storage.

This procedure includes steps for configuring a credential wallet to store TDE master keys, because backups are encrypted before they are archived to a cloud repository. The initial configuration tasks are performed in the Oracle Key Vault to prepare the wallet. RACLI commands were developed to assist configuring the Recovery Appliance for archive-to-cloud and using the wallet. At the end, a job template is created and run for archive-to-cloud.

> **Note**
>
> When the backups are created as *space-efficient backups* (using compression and TDE encryption), their restoration requires the Recovery Appliance, because the database communicates with the appliance to process space-efficient backups. As such, these backups cannot be restored directly from the media.

> **Note**
>
> Best practics for key and wallet management.
>
> - Database must be configured with TDE Encryption/Wallet (File or OKV).
>   - CDB and every PDB must have encryption key enabled.
>   - RMAN password-based encryption not supported (must use auto-login wallet)
> - File based wallet / OKV must be backed up separately. The wallets and keys are not allowed to be stored on the same system as backups
> - No keys should ever be removed from the wallet, because virtual fulls may contains blocks from L0/L1s that were taken when different master keys were in effect.

11.1 Grouping Backup Pieces

The performance of copy-to-tape and archive-to-cloud is improved by grouping archived logs from protected databases' real-time redo into fewer number of backup sets.

Protected databases can achieve real-time protection by enabling real-time redo transport to the Recovery Appliance. Each received redo log on the appliance is compressed and written to the storage location as an individual archived log backup. These log backups can be archived to tape to support fulls and incremental backups that are archived for long-term retention needs.

Inter-job latencies can happen between writing each backup piece during copy-to-tape operations. When the number of backup pieces is high, this pause constitutes a large

Copyright © 2014, 2026, Oracle and/or its affiliates.

percentage of the time the tape drive is unavailable. This means five (5) 10GB pieces will go to tape more quickly than fifty (50) 1GB pieces.

Recovery Appliance addresses inter-job latency by grouping the archived log backup pieces together and copying them as a single backup piece. Therefore this results in larger backup pieces on tape storage than previous releases. This feature is enabled by default. `DMBS_RA CONFIG` has the parameter `group_log_max_count` for setting the maximum archived logs per backup piece that is copied to tape; its default is 1. The `group_log_backup_size_gb` parameter is used to limit the size of these larger backup pieces; its default is 256 GB.

> **Note**
>
> Storing backups in the cloud does not have the latency of tape devices. The number of streams that can be assigned in the cloud is not limited by the (tape) device. Grouping archived log files is recommended for tape operations, but not necessarily cloud.

11.2 Pre-requisites for Archive-to-Cloud

The following prerequisites must be met before starting to use cloud storage with the Recovery Appliance.

- Protected database(s) should already be enrolled and backups taken to the Recovery Appliance.

  This is covered in Configuring Recovery Appliance for Protected Database Access . Brief review:

  - Create a virtual private catalog user.
  - Enroll the protected database.
  - Update the properties for the protected database.
- The Recovery Appliance has been registered and enrolled at an Oracle Key Vault.

  OKV is not required if space-efficient encrypted backups is enabled, the log destination to Recovery Appliance is set for encryption enabled, and `SECURE_MODE` is enabled in the protection policy. Refer to Configuring Encrypted SBT for more information.
- Archive-to-cloud features are only supported on small endian databases. Only Linux and Windows.

  Big endian databases that attempt to archive-to-cloud cause error ORA-64800: *unable to create encrypted backup for big endian platform*, because the operation cannot create encrypted backup for big endian platforms.

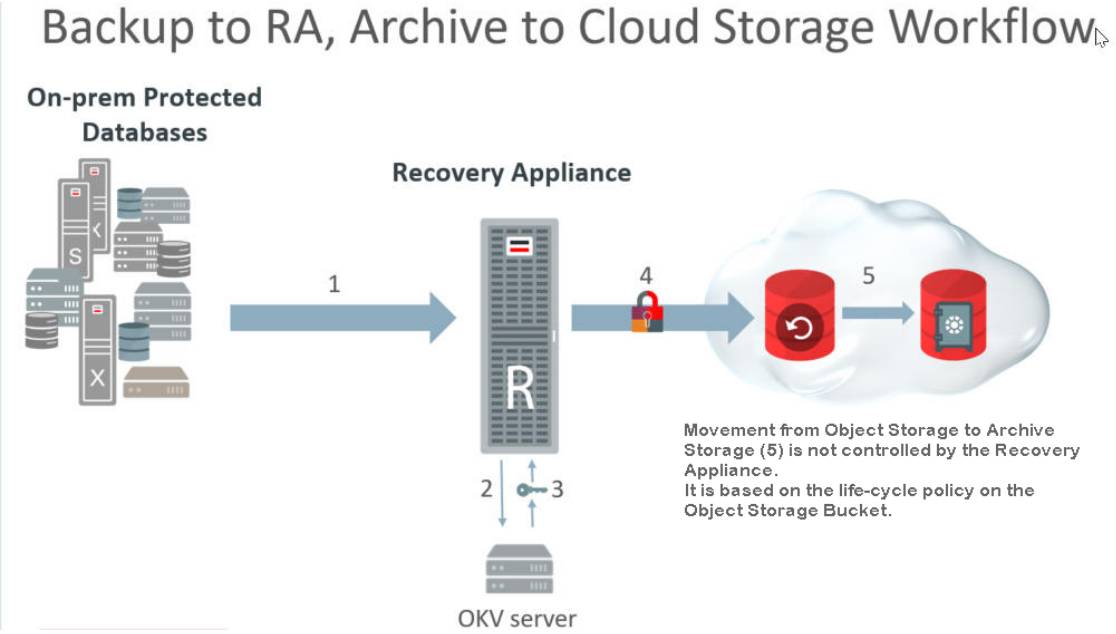
11.3 Flow for Archive-to-Cloud Storage

All backup objects archived to cloud storage are encrypted using a random Data Encryption Key (DEK). A Transparent Data Encryption (TDE) master key for each protected database is used to encrypt the DEK; the encrypted DEK is stored in the backup piece. The Oracle Key Vault (OKV) contains the TDE master keys; it does not contain the individual DEKs used to encrypt backups written to tape or cloud. A protected database may acquire many TDE master keys with time, so restoration of an individual archived object requires the protected database's master key in use at time of backup.

The following image shows the flow for backing up to a Recovery Appliance that archives to cloud storage. The restore operations are predicated on this backup and archive flow.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Figure 11-1 Flow for Backups to Cloud Storage



1. Incremental backups of the database are performed regularly to the Recovery Appliance. This happens at a different interval than the following archive operations.
2. When the scheduled archive-to-cloud operation starts, the Recovery Appliance requests a master key for the protected database from the OKV Server.
3. The OKV returns the protected database's master key. If one doesn't exist for the protected database, a new master key is generated. (A new master key can be generated whenever desired.)
   a. A DEK is generated for the backup object(s).
   b. The backup objects are encrypted using the DEK.
   c. Using the master key, the Recovery Appliance encrypts the DEK and stores this with the backup object.
4. The life-cycle policy for a given database determines if and when its backup objects are written to tape or cloud storage.
5. The life-cycle policy of the object storage bucket determines if and when a backup object in cloud storage moves from object storage to archive storage. The Recovery Appliance does not control this.

11.4 Oracle Key Vault and Recovery Appliance

The Oracle Key Vault (OKV) stores the TDE master keys and also keeps track of all enrolled endpoints.

Endpoints are the database servers, application servers, and computer systems where actual cryptographic operations such as encryption or decryption are performed. Endpoints request OKV to store and retrieve security objects.

A brief overview of the Oracle Key Vault (OKV) configurations:

Copyright © 2014, 2026, Oracle and/or its affiliates.

- All compute nodes of the Recovery Appliance are registered and enrolled as OKV endpoints.
- A single OKV endpoint group contains all the endpoints corresponding to all of the compute nodes of the Recovery Appliance.
- A single wallet is shared and configured as `'Default Wallet'` for all endpoints corresponding to all of the compute nodes of the Recovery Appliance.
- The OKV endpoint group is configured with read/write/manage access to the shared virtual wallet.
- If more than one Recovery Appliance is involved, each Recovery Appliance has its own end point group and wallet.
- The host-specific `okvclient.jar` is created and saved during the enrollment process of each endpoint to the staging path on its respective node. If the root user is performing the operation, the `/radump` is the staging path. If a named user (such as `raadmin`) is performing the operation, then the staging has to be in `/tmp`. The staged file has to be named either as-is `okvclient.jar` or `<myHost>-okvclient.jar`, where `<myHost>` matches what `hostname` returns.

> **Note**
>
> Refer to *Oracle Key Vault Administrator's Guide* for more information.
>
> - Overview of Oracle Key Vault Concepts
> - Managing Oracle Key Vault Endpoints

11.4.1 Review: Oracle Key Vault

This reference section employs concepts from the *Oracle Key Vault Administrator's Guide* (OKV).

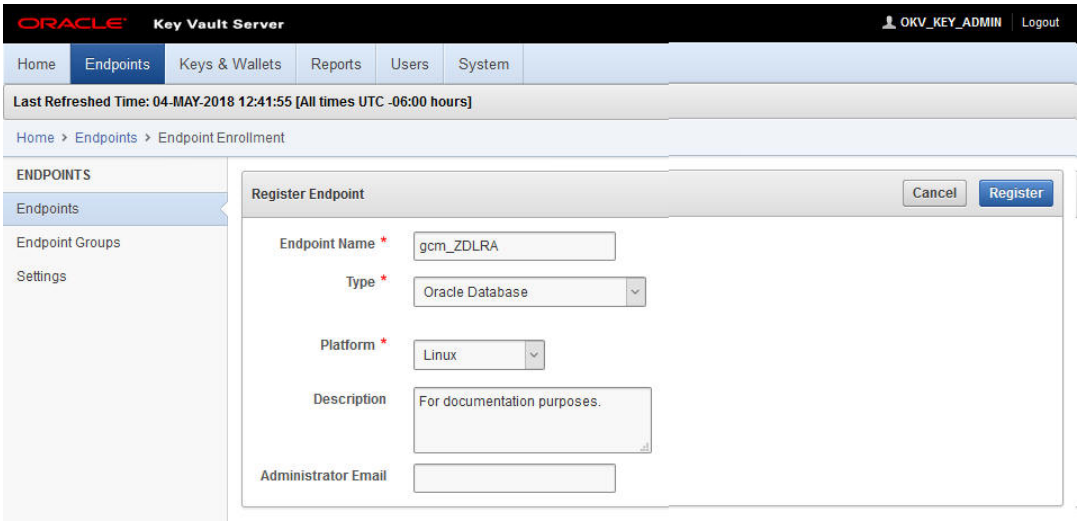
The OKV administrator performs these tasks, and are a pre-requisite for the operations performed by the Recovery Appliance administrator. The OKV administrator configures the OKV Endpoints.

11.4.2 Creating the Endpoints

These operations for created an **Endpoint** are performed from the Key Vault Server Web Console.

1. Log into the Oracle Key Vault Server.
2. Click **Endpoints** tab.
3. Click **Add** button in right corner of the **Endpoints** page.
4. Enter the information specific to the Recovery Appliance node that the endpoint is to be associated with. (Name/Type/Platform/Desc/Email)

Copyright © 2014, 2026, Oracle and/or its affiliates.

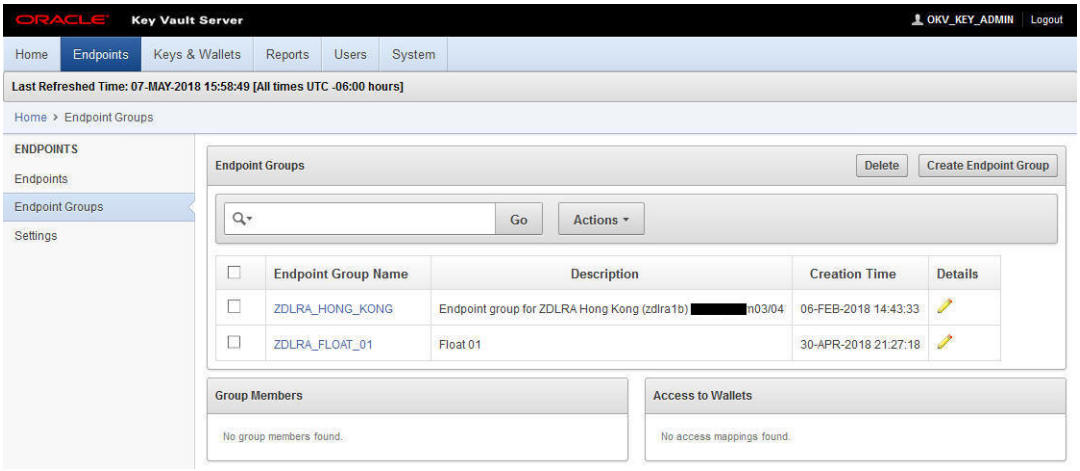


5. Click **Register** button on the right.
6. Repeat the above steps to create an endpoint for every Recovery Appliance node.

11.4.3 Creating the Endpoint Group

These operations for creating an **Endpoint Group** are performed from the Key Vault Server Web Console.

1. Click **Endpoints** tab.
2. Click **Endpoint Groups** option on the left.



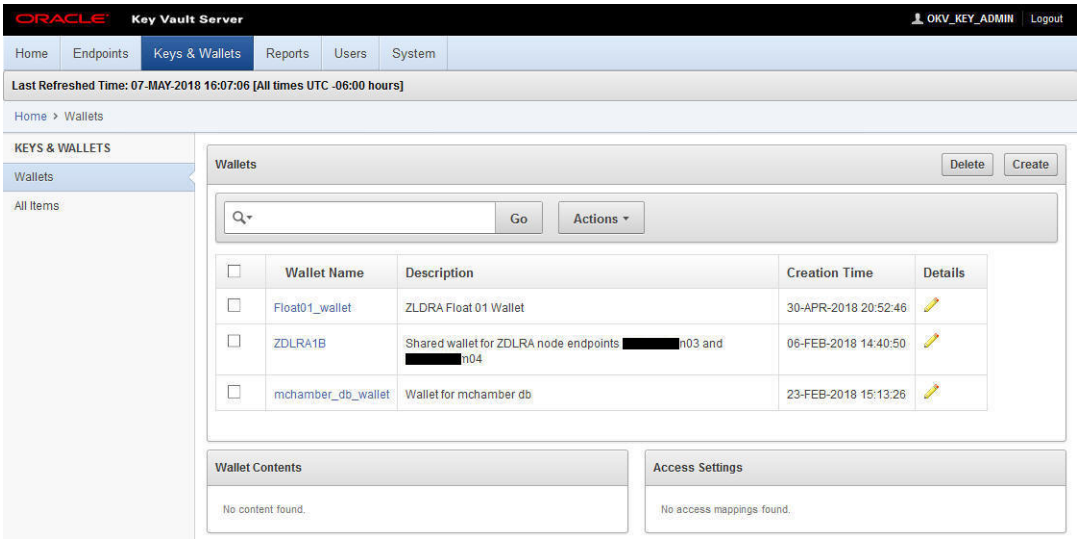
3. Click **Create Endpoint Group** button on top right.
4. Enter name and description, and select all endpoints created in the previous operations.
5. Click **Save** button on the right.

11.4.4 Creating a Wallet

These operations for created an **Wallet** are performed from the Key Vault Server Web Console.

1. Click **Keys & Wallets** tab.

Copyright © 2014, 2026, Oracle and/or its affiliates.

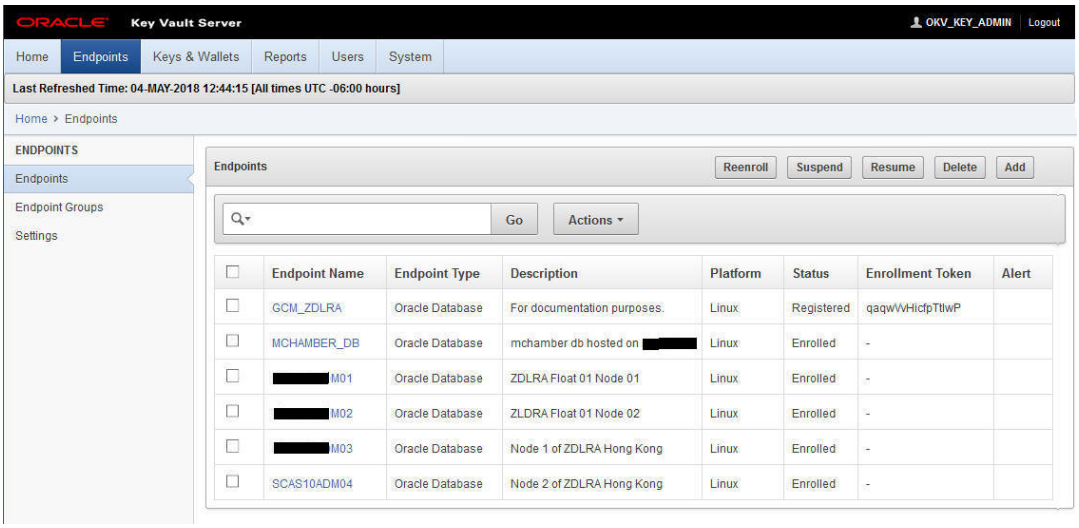


2. Click **Create** button at top right.
3. Enter name and description specific to the first node/endpoint.
4. Click on the **Save** button on the right.

11.4.5 Associating Default Wallet with Endpoints

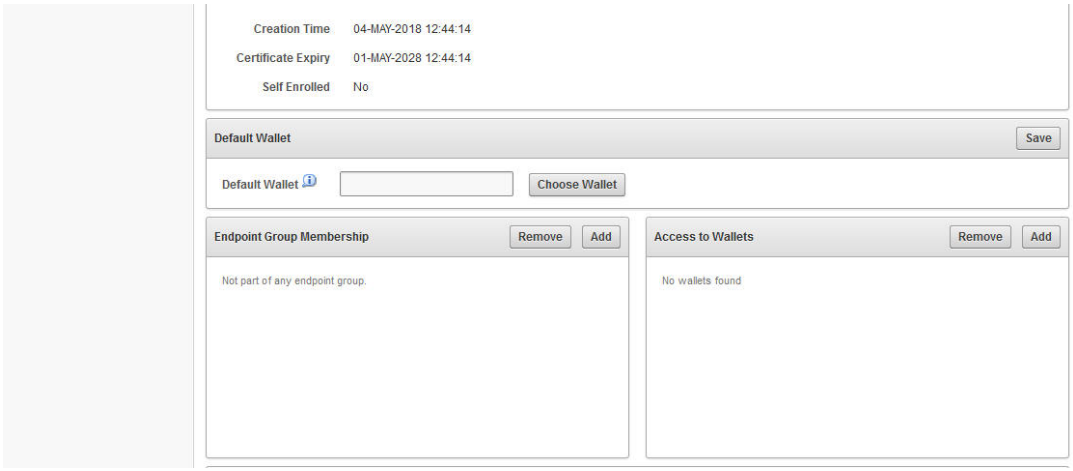
These operations for associating the virtual wallet with an **Endpoint** are performed from the Key Vault Server Web Console.

1. Click **Endpoints** tab.



2. Click on the specific name for the endpoint being associated with a wallet.

Copyright © 2014, 2026, Oracle and/or its affiliates.



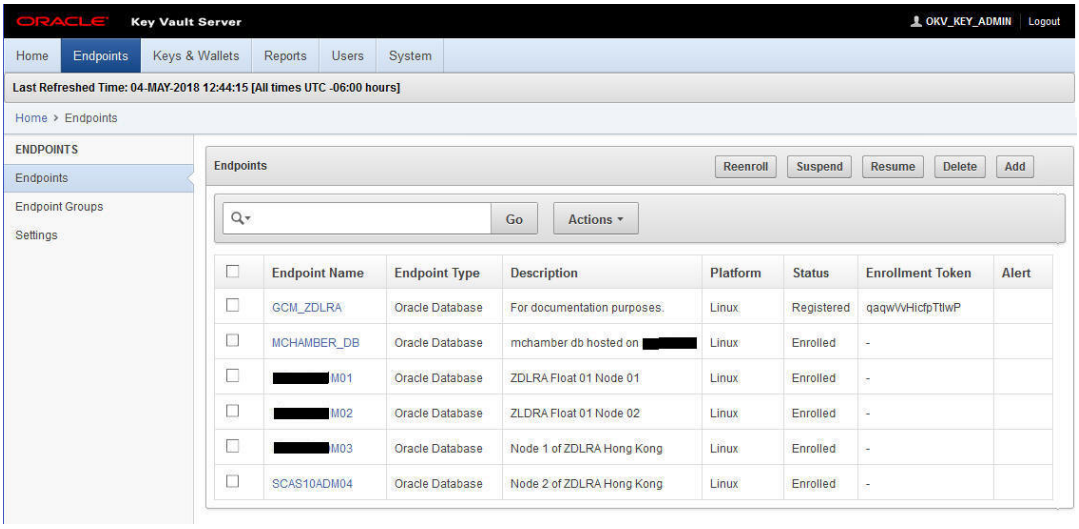
3. In **Default Wallet** section, click **Choose Wallet** button.
4. Click on the name of the wallet created above, and click **Select** to assign endpoints.
5. Click **Save** button on the right.
6. Repeat wallet assignment for other endpoints. The same wallet is assigned to those endpoints.

11.4.6 Acquiring the Enrollment Tokens

These operations for acquiring the enrollment tokens are performed from the Key Vault Server Web Console.

1. Click **Endpoints** tab.

   The page now includes enrollment tokens specific to each endpoint/node.



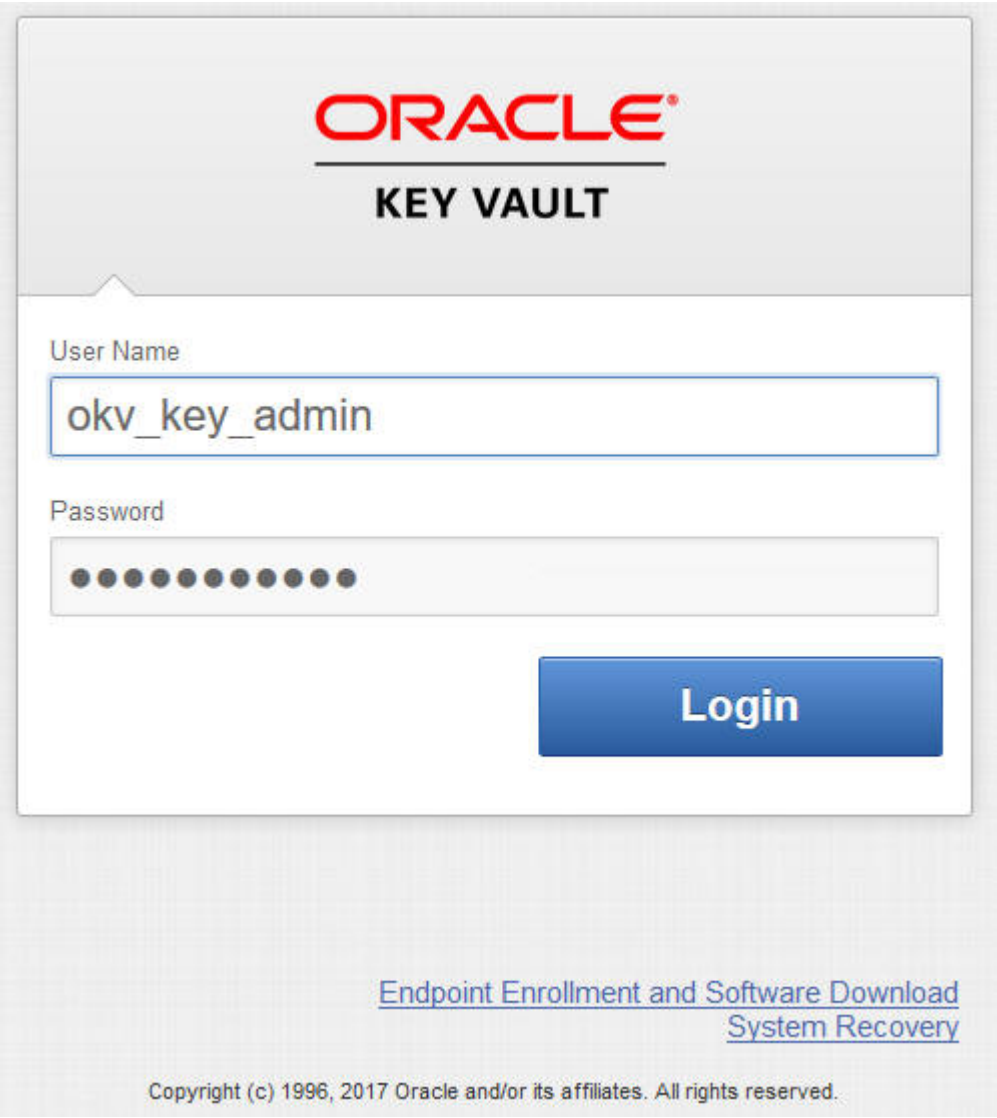
2. Copy and retain (in a file) the enrollment token specific to each endpoint, because it is used in a later enrollment step.
3. Logout of the web interface. This step is required in order for other steps to display refreshed information.

Copyright © 2014, 2026, Oracle and/or its affiliates.

11.4.7 Downloading the OKV Client Software

These operations for downloading the OKV client software are performed from the Key Vault Server Web Console.

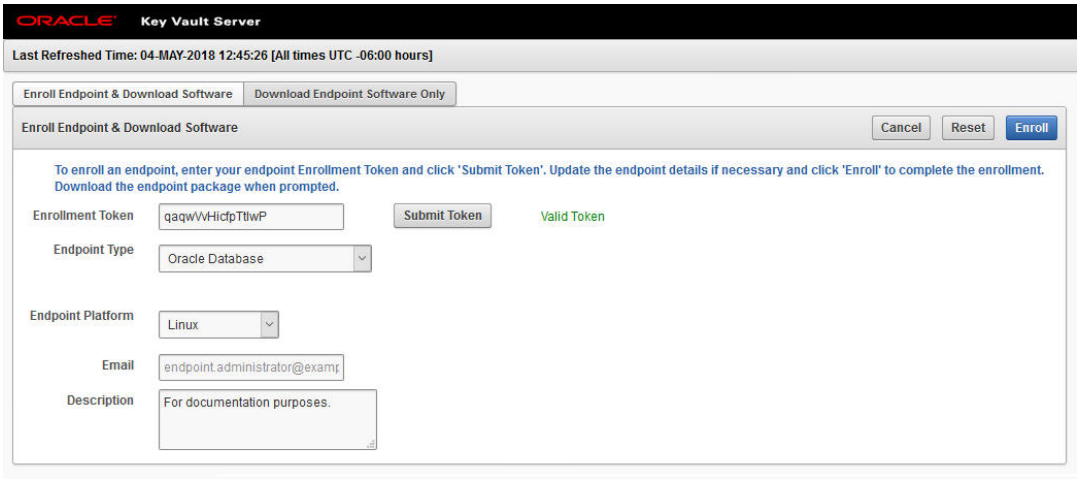
The follow steps are repeated for each node of the Recovery Appliance. These steps download JAR files that are specific to the Recovery Appliance node.



1. Click on **Endpoint Enrollment and Software Download** link on the Management Console. This link is below the login section.
2. Using an enrollment token saved from previous steps, paste it into the **Enrollment Token** field.
3. Click **Submit Token** button.

Copyright © 2014, 2026, Oracle and/or its affiliates.

This displays the endpoint information entered.



4. Click **Enroll** at top right. A progress bar appears with the text *"processing"*.

   A software download window appears after the request has been processed.

5. The `okvclient.jar` should be re-named in a host-specific manner, saved locally, and then copied to the `/radump` directory on its respective Recovery Appliance node.

   The staged file in the `/radump` has to be named either:

   - as-is `okvclient.jar`, or
   - `"<myHost>-okvclient.jar"`, where `<myHost>` matches what `hostname -s` returns.

   Renaming the file `<myHost>-okvclient.jar` avoids confusion and the temptation to use an `okvclient.jar` on any other node but the one it was generated for.

6. Repeat the above steps for each node of the Recovery Appliance.

> **Note**
>
> Do ***not*** install the JAR files at this point in time. Installation happens after other Recovery Appliance configuration steps.
>
> The JAR files are only valid until enrollment of OKV endpoints are complete.

11.5 Recovery Appliance Cloud Archive Configuration

This section configures the Recovery Appliance to be able to use wallets and cloud objects as required for archive-to-cloud.

11.5.1 Configuring the Credential Wallet and Encryption Keystore

All database backup pieces are DEK encrypted before any copy-to-tape or archive-to-cloud operation.

These steps create a shared wallet to be used by all nodes of the Recovery Appliance. The wallet stores TDE master keys that encrypt the individual DEK keys.

Copyright © 2014, 2026, Oracle and/or its affiliates.

1. Create the Recovery Appliance credential wallet. You are prompted to enter new passwords for the keystore and then the wallet. The credentials to access the Recovery Appliance encryption keystore are saved in this wallet.

```
[root@myComputeNodeX ~]# racli add credential_wallet

Fri Jan 1 08:56:27 2018: Start: Add Credential Wallet
Enter New Keystore Password: <OKV_endpoint_password>
Confirm New Keystore Password:
Enter New Wallet Password: <ZDLRA_credential_wallet_password>
Confirm New Wallet Password:
Re-Enter New Wallet Password:
Fri Jan 1 08:56:40 2018: End: Add Credential Wallet
```

   For details on the command options, refer to "racli add credential_wallet".

2. Configure the Recovery Appliance encryption keystore. This keystore contains one or more TDE Master keys for each Recovery Appliance client database, plus the Recovery Appliance's TDE Master key. The per-client TDE Master keys are used to encrypt backups pieces that are copied to the cloud.

   NOT_SUPPORTED

   The Recovery Appliance database is restarted to activate the keystore; plan for short outage.

```
[root@myComputeNodeX ~]# racli add keystore --type hsm --restart_db

Updating log /opt/oracle.RecoveryAppliance/log/racli.log
Fri Jan 1 08:57:03 2018: Start: Configure Wallets
Fri Jan 1 08:57:04 2018: End: Configure Wallets
Fri Jan 1 08:57:04 2018: Start: Stop Listeners, and Database
Fri Jan 1 08:59:26 2018: End: Stop Listeners, and Database
Fri Jan 1 08:59:26 2018: Start: Start Listeners, and Database
Fri Jan 1 09:02:16 2018: End: Start Listeners, and Database
```

   For details on the command options, refer to "racli add keystore".

A shared wallet is created that all nodes of the Recovery Appliance use. It stores TDE master keys that encrypt the individual DEK keys.

11.5.2 Installing the OKV Client Software

Each node of the Recovery Appliance needs to have the appropriate client software for the Oracle Key Vault (OKV). This is accomplished using RACLI in one step.

If the user is not an `admin_user` or `root` user who have access to `/radump`, then stage the `okvclient.jar` file in `/tmp` on both nodes.

1. From the primary node of Recovery Appliance, run only once the following command. It adds all OKV endpoints associated with the Recovery Appliance. It applies to ~*all*~ nodes.

```
[root@myComputeNodeX ~]# racli install okv_endpoint
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
Wed August 23 20:14:40 2018: Start: Install OKV End Point [node01]
Wed August 23 20:14:43 2018: End: Install OKV End Point [node01]
Wed August 23 20:14:43 2018: Start: Install OKV End Point [node02]
Wed August 23 20:14:45 2018: End: Install OKV End Point [node02]
```

For details on the command options, refer to "racli install okv_endpoint".

2. Verify that the Oracle Key Vault endpoint software has been provisioned properly.

```
[root@myComputeNodeX ~]# racli status okv_endpoint

Node: node02
Endpoint: Online
Node: node01
Endpoint: Online
```

All nodes should have the client software for the OKV.

11.5.3 Enabling the Encryption Keystore and Creating a TDE Master Key

This task enables a keystore and creates the first TDE master key.

The OKV endpoint keystore is also known as the *"OKV shared wallet."* Once a keystore has been created, it must be enabled for use and the first TDE master key created for it.

1. Open the keystore so that it can be used.

```
[root@myComputeNodeX ~]# racli enable keystore
```

For details on the command options, refer to "racli enable keystore".

2. Create a TDE master key for the Recovery Appliance.

```
[root@myComputeNodeX ~]# racli alter keystore --initialize_key
```

11.5.4 Creating Cloud Objects for Archive-to-Cloud

This task creates the OCI objects `Cloud_Key` and `Cloud_User` for use with archive-to-cloud.

1. Add a `Cloud_Key`. This object is specific for OCI Cloud Archive support.

```
[root@myComputeNodeX ~]# racli add cloud_key --key_name=example_key

Thu Sep  1 18:11:23 2022: Using log file
/opt/oracle.RecoveryAppliance/log/racli.log
Thu Sep  1 18:11:23 2022: Start: Add Cloud Key example_key
Thu Sep  1 18:11:25 2022: Start: Creating New Keys
Thu Sep  1 18:11:25 2022: Oracle Database Cloud Backup Module Install Tool,
build 19.9.0.0.0DBBKPCSBP_2022-05-02
Thu Sep  1 18:11:25 2022: OCI API signing keys are created:
Thu Sep  1 18:11:25 2022:   PRIVATE KEY -->
/raacfs/raadmin/cloud/key/example_key/oci_pvt
Thu Sep  1 18:11:25 2022:   PUBLIC  KEY -->
/raacfs/raadmin/cloud/key/example_key/oci_pub
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
Thu Sep  1 18:11:25 2022: Please upload the public key in the OCI console.
Thu Sep  1 18:11:25 2022: End: Creating New Keys
Thu Sep  1 18:11:26 2022: End: Add Cloud Key example_key
```

For details on the command options, refer to "racli add cloud_key".

2. Open the OCI console, and sign in. The console is located at `https://console.<region>.oraclecloud.com`. If you don't have a login and password for the Console, contact an administrator.
3. From the OCI console, acquire the key's fingerprint.
   a. View the details for the user who will be calling the API with the key pair.
      - If you're signed in as this user, click your username in the top-right corner of the Console, and then click **User Settings**.
      - If you're an administrator doing this for another user, instead click **Identity**, click **Users**, and then select the user from the list.
   b. Click **Add Public Key**.
   c. Paste the contents of the PEM public key in the dialog box and click **Add**.
   d. **Important:** Copy the key's fingerprint, because it is needed in later steps.

   The key's fingerprint is displayed (for example, 12:34:56:78:90:ab:cd:ef:12:34:56:78:90:ab:cd:ef).
4. (Optional) After you've uploaded your first public key, you can upload additional keys. You can have up to three API key pairs per user. In an API request, you specify the key's fingerprint to indicate which key you're using to sign the request.
5. Modify `Cloud_Key` by adding the fingerprint.

```
[root@myComputeNodeX ~]# racli alter cloud_key
--key_name=example_key
--fingerprint=12:34:56:78:90:ab:cd:ef:12:34:56:78:90:ab:cd:ef

Tue Jul  2 05:40:06 2019:   Start: Alter Cloud Key example_key
Tue Jul  2 05:40:08 2019:   End: Alter Cloud Key example_key
```

6. Add `Cloud_User` object.

```
[root@myComputeNodeX ~]# racli add cloud_user
--user_name=sample_user
--key_name=example_key
--user_ocid=ocid1.user.oc1..abcedfghijklmnopqrstuvwxyz0124567901
--tenancy_ocid=ocid1.tenancy.oc1..abcedfghijklmnopqrstuvwxyz0124567902
--
compartment_ocid=ocid1.compartment.oc1..abcedfghijklmnopqrstuvwxyz012456790
3

Tue Jun 18 13:28:45 2019: Using log file /opt/oracle.RecoveryAppliance/log/
racli.log
Tue Jun 18 13:28:45 2019: Start: Add Cloud User sample_user
Tue Jun 18 13:28:46 2019: End: Add Cloud User sample_user
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

--user_name

The name to be associated with this particular cloud user. This is a logical name for the Recovery Appliance; it will be used in the Recovery Appliance `cloud_location`. It does not have to match the actual ZFS user name

--key_name

The specific cloud key to be associated with this cloud user. This is the `cloud_key` object created in step #1.

--tenancy_ocid

The tenancy OCID for the Oracle Bare Metal Cloud account. This is the value to be used and does not change.

--user_ocid

The user OCID for the Oracle Bare Metal Cloud account. This is the OCID for the object storage user on the ZFS. It is always in the form `ocid1.user.oc1..<zfs_username>`.

--compartment_ocid

The compartment OCID within the tenancy of the Oracle Bare Metal Cloud Account. The compartment OCID is always the ZFS share name.

For details on the command options, refer to "racli add cloud_user".

7. Verify `Cloud_User` was created by listing it.

```
[root@myComputeNodeX ~]# racli list cloud_user --user_name=sample_user

Tue Jul  2 06:45:13 2019: Using log file /opt/oracle.RecoveryAppliance/log/
racli.log
Tue Jul  2 06:45:13 2019: Start: List Cloud User
              Cloud User:  sample_user
               User Name: sample_user
                 User ID: 3
                User OCID:
ocid1.user.oc1..abcedfghijklmnopqrstuvwxyz0124567901
            Tenancy OCID:
ocid1.tenancy.oc1..abcedfghijklmnopqrstuvwxyz0124567902
        Compartment OCID:
ocid1.compartment.oc1..abcedfghijklmnopqrstuvwxyz0124567903
          Cloud Key Name: hk_key_1

Tue Jul  2 06:45:14 2019: End: List Cloud User
```

11.5.5 Adding Cloud Location

This task configures a cloud bucket location for archive-to-cloud.

Creation of a `cloud_location` requires that a `cloud_user` object has already been created. Each `cloud_location` creation is tied to a singular, specified `cloud_user`. Resulting object name translates to cloud `sbt_library` name, such as `bucket_cloud_user`. In this model, each cloud location is one-to-one `cloud_user` to `cloud_location`.

The options given to RACLI are passed to the installer, which handles setting lifecycle management for the bucket.

Copyright © 2014, 2026, Oracle and/or its affiliates.

When completed, Object Storage is authorized to move backups to Archive Storage, as per Configuring Automatic Archival to Oracle Cloud Infrastructure.

1. Add cloud location to the Recovery Appliance. This creates a `sbt_library` for archive-to-cloud.

```
[root@myComputeNodeX ~]# racli add cloud_location

--cloud_user=CLOUD_USER_NAME
--host=HOST_URL
--bucket=OCI_BUCKET_NAME
[--enable_archive |  --disable_archive]
[--archive_after_backup=NUMBER:{DAYS|YEARS}  --streams=NUMBER --
proxy_host=HTTP_SERVER
--proxy_port=HTTP_PORT  --proxy_id=HTTP_USER --proxy_pass=HTTP_PASS
--import_all_trustcert=X509_CERT_PATH  --retain_after_restore=NUMBER:HOURS]
[-guaranteed={yes|no}]
[--immutable
--temp_metadata_bucket=METADATA_BUCKET_NAME]


Tue Jun 18 13:30:51 2019: Using log file /opt/oracle.RecoveryAppliance/log/
racli.log
Tue Jun 18 13:30:51 2019: Start: Add Cloud Location
<OCI_BUCKET_NAME>_<CLOUD_USER_NAME>
Tue Jun 18 13:30:57 2019: End: Add Cloud Location
<OCI_BUCKET_NAME>_<CLOUD_USER_NAME>
```

`--bucket`

The name of the bucket where the backup will go. Note that the install tool will create the specified bucket if it does not exist.

The bucket name is the directory which will be created in the `--compartment_ocid` ZFS share in step #2.

Bucket names are case sensitive, allowed characters are alphanumeric characters, /, -, _ and period (.), other special characters are not allowed. Bucket name max length is 255 characters (one less than OCI 256).

`--cloud_user`

Previously configured `cloud_user` object with all authentication requirements. This is the same logical name used for the `cloud_user` creation in step #2

`--host`

Host name for the Oracle Bare Metal Cloud account. This is the ZFS hostname or IP address always followed by /oci - Do not use https.

`--streams`

The maximum number of streams used during data send/receive operations between the ZFS and Recovery Appliance. The specific stream count will be configured when defining the copy job template in a later step below. It is not recommended to exceed 256 total open connections to Object Storage on a single ZFS appliance.

- Just like OCI public cloud buckets, the `cloud_location` will be used as a Media Management Library (MML) in the ZDLRA. The MML will appear as `<bucket_name>_<user_name>`.

Copyright © 2014, 2026, Oracle and/or its affiliates.

- Attribute sets will be created on the Recovery Appliance based on the number of `--steams` specified above

> **Note**
>
> Validating that the cloud object was created properly is critical. If `--enable_archive=TRUE` (listed as `Archive: TRUE`), the cloud bucket can perform archive-to-cloud operations. If `--enable_archive` is not provided, the default is `FALSE`, which means the created cloud location cannot perform archive-to-cloud operations and becomes cold storage.

2. List `cloud_location` object(s) to verify they were created correctly.

```
[root@myComputeNodeX ~]# racli list cloud_location --
location_name=<CLOUD_LOCATION_NAME>
Fri Oct 25 06:27:18 2019: Using log file /opt/oracle.RecoveryAppliance/log/
racli.log
Fri Oct 25 06:27:18 2019: Start: List Cloud Location
Cloud Location <CLOUD_LOCATION_NAME>
           Location Name: <CLOUD_LOCATION_NAME>
                 Archive: TRUE
    Archive After Backup: 7:Days
                    Host: https://<HOST_URL>
                  Bucket: <OCI_BUCKET_NAME>
             Location ID: 21
              Proxy Host: 127.0.0.1
              Proxy Port: 80
    Retain After Restore: 1:Hours
                 Streams: 6
                 User ID: 1
             SBT Library: <CLOUD_LOCATION_NAME>
      Attribute Set Name: <CLOUD_LOCATION_NAME>_1
           Backup Stream: 1
      Attribute Set Name: <CLOUD_LOCATION_NAME>_2
           Backup Stream: 2
      Attribute Set Name: <CLOUD_LOCATION_NAME>_3
           Backup Stream: 3
      Attribute Set Name: <CLOUD_LOCATION_NAME>_4
           Backup Stream: 4
      Attribute Set Name: <CLOUD_LOCATION_NAME>_5
           Backup Stream: 5
      Attribute Set Name: <CLOUD_LOCATION_NAME>_6
           Backup Stream: 6
Fri Oct 25 06:27:18 2019: End: List Cloud Location
```

If the cloud location was created improperly based on this verification step, use `racli remove cloud_location` and run with the correct arguments `racli add cloud_location`.

In later steps, you need the name of the attribute set to create a `sbt_job_template`. This can be derived from the "`racli list cloud_location --long`" output. The SBT library and attribute sets created by `racli` can be displayed using `dbms_ra`, but should not be modified.

Copyright © 2014, 2026, Oracle and/or its affiliates.

11.5.6 Adding an Immutable Cloud Location

This task configures an immutable cloud bucket location for archive-to-cloud.

An immutable bucket is one that retains backups in cloud storage for a period specified by the `KEEP UNTIL` attribute of the backup. An immutable cloud location requires two buckets that must be created in advance using the OCI Console, the ZFS console, or the OCI command line interface. The cloud buckets are:

- **Regulatory Compliance Bucket** has retention rule set and locked.
- **Temporary Metadata Bucket** without retention rules.

The retention rules apply to the whole bucket. Therefore, it should not use automatic lifecycle rules triggering `Delete`. The recommendation is one database per immutable cloud location.

1. Configure the database client with a Recovery Appliance and take a backup on the client.
2. Install OKV endpoint on the Recovery Appliance.
3. Create two (2) standard buckets using the OCI console or OCI-CLI.

   The target bucket is used to store the backups. The second bucket is used for temporary metadata during backup operations.
4. Using the OCI console or OCI-CLI, create a retention rule on the target bucket to make content immutable for the desired period of time.

   No retention rule is required to be set on the temporary metadata bucket.
5. Use RACLI to create a `cloud_location` using these buckets specifying the following parameters

   - `--immutable`
   - `--bucket` <target bucket with retention rule>
   - `--temp_metadata_bucket` <temporary metadata bucket>

   To store immutable backups in the object storage archive tier, also use these parameters.

   - `--enable_archive=true`
   - `--archive_after_backup=1:DAYS`
   - `--retain_after_restore=24:HOURS`

```
[root@myComputeNodeX ~]# racli add cloud_location
--cloud_user=<CLOUD_USER_NAME>
--host=https://<OPC_STORAGE_LOCATION>
--bucket=<OCI_BUCKET_NAME>
--proxy_port=<HOST_PORT>
--proxy_host=<PROXY_URL>
--proxy_id=<PROXY_ID>
--proxy_pass=<PROXY_PASS>
--streams=<NUM_STREAMS>
[--enable_archive=TRUE]
--archive_after_backup=<number>:[YEARS | DAYS]
[--retain_after_restore=<number_hours>:HOURS]
--import_all_trustcert=<X509_CERT_PATH>
--immutable
--temp_metadata_bucket=<metadata_bucket>
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
[--enable_archive=true --archive_after_backup=2:DAYS --
retain_after_restore=8:HOURS]
```

6. Create SBT_JOB_TEMPLATE for archive to cloud.

11.6 Configuring the ZFS OCI Object Storage as a Cloud Repository

Use the ZDLRA archive-to-cloud to copy or move long term retention backups from the Recovery Appliance to the ZFS OCI object storage.

This configuration has the following pre-requisites:

- The ZFS Appliance must have the OCI Object Storage interface enabled as documented in Oracle ZFS Storage Appliance Object API Guide for Oracle Cloud Infrastructure Object Storage Service Support. There is no Archive Storage tier for ZFS, so only Standard Object Storage is supported.
- The Recovery Appliance must be configured to use Oracle Key Vault. This must be done, because the Recovery Appliance is not aware that the target is a local ZFS appliance and treats the operation the same as an OCI object storage in the Oracle cloud. When moving data to what appears to be the cloud, the Recovery Appliance must encrypt the data before sending it to the ZFS OCI object storage. An Oracle Key Vault (OKV) server must be used to manage the keys for this encryption process.

After configuring and enabling OCI Object Storage on the ZFS appliance and setting up OKV usage with the Recovery Appliance, these are the steps to configure the ZFS bucket as a `cloud_location`. This section highlights the difference in the configuration process for use with ZFS.

1. Create the cloud_key on the Recovery Appliance with `racli add cloud_key`.
   - If a new key pair is being created, the public key must be copied to the ZFS appliance.
   - If an existing key pair is being used for which the public key is already on the ZFS appliance, the file containing the matching private key must be specified
   - Update the `cloud_key` to add the fingerprint provided on the ZFS interface using the `racli alter cloud_key` command on Recovery Appliance.
2. Create the `cloud_user` on the Recovery Appliance using the `racli add cloud_user` command.
3. Create the `cloud_location` connection using the `racli add cloud_location` command.
4. Create an SBT job template and schedule on the Recovery Appliance.

The Recovery Appliance sends now long term retention backups to the ZFS OCI Object Storage bucket, just as it would for Object Storge in Oracle Cloud Infrastructure cloud.

11.7 Creating a Job Template

This task creates a job template for archive-to-cloud.

The options given to RACLI are passed to the installer, which handles setting lifecycle management for the bucket.

1. Log in to SQL*Plus with an `admin db_user` user.

Copyright © 2014, 2026, Oracle and/or its affiliates.

2. Create a `SBT_JOB_TEMPLATE` for archive-to-cloud. The supported algorithms are 'AES128', 'AES192', and 'AES256'. The attribute set name is either `<CLOUD_LOCATION_NAME>_1` for 1 stream or `<CLOUD_LOCATION_NAME>_2` for two (2) streams in parallel.

```
SQL> exec
dbms_ra.create_sbt_job_template(template_name=>'<COPY_TO_CLOUD_TEMPLATE_NAM
E>',
protection_policy_name=>'BRONZE',
attribute_set_name=>'< Attribute Set Name >',
backup_type=>'ALL',
full_template_name=>'<COPY_TO_CLOUD_TEMPLATE_NAME>',
from_tag=>NULL,
priority=>100,
copies=>1,
window=>NULL,
compression_algorithm=>'<SUPPORTED_COMPRESSION>',
encryption_algorithm=>'<SUPPORTED_ALGO>');
```

> **Note**
>
> When using compliance, Oracle recommends having one bucket per database. When you create a job template in a compliance environment, use `db_unique_name` instead of `protection_policy_name` in your job template, unless the protection policy is used by a single database.

```
SQL> exec
dbms_ra.create_sbt_job_template(template_name=>'<COPY_TO_CLOUD_TEMPLATE_NAM
E>',
db_unique_name=> '< Database Name>',
attribute_set_name=>'< Attribute Set Name >',
backup_type=>'ALL',
full_template_name=>'<COPY_TO_CLOUD_TEMPLATE_NAME>',
from_tag=>NULL,
priority=>100,
copies=>1,
window=>NULL,
compression_algorithm=>'<SUPPORTED_COMPRESSION>',
encryption_algorithm=>'<SUPPORTED_ALGO>');
```

For details on the command options, refer to "CREATE_SBT_JOB_TEMPLATE".

3. Run the archive-to-cloud job..

```
SQL> exec dbms_ra.queue_sbt_backup_task('<COPY_TO_CLOUD_TEMPLATE_NAME>');

PL/SQL procedure successfully completed.
```

4. Verify backup initiation.

```
SQL> SELECT task_type, state, TRUNC(last_execute_time), COUNT(*)
FROM ra_task
WHERE state IN ('RUNNING','EXECUTABLE','WAITING','LIBRARY_WAIT')
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
AND archived = 'N'
GROUP BY task_type, state, TRUNC(last_execute_time);


TASK_TYPE      STATE         TRUNC(LAS  COUNT(*)
-------------- ------------- ---------  ----------
BACKUP_SBT     EXECUTABLE    18-JUN-18  18
BACKUP_SBT     RUNNING       18-JUN-18  2
```

11.8 Creating or Re-Creating Protected Database TDE Master Keys

This step creates or recreates the TDE master keys used from that point forward for encrypting the DEK keys used on protected databases.

Security policies specify the frequency or circumstances for the creation of new TDE master keys for protected databases. This operation is called "re-key", and is performed as user *rasys* in PL/SQL on the Recovery Appliance.

The following re-key options are available.

- Re-key ~*all*~ protected databases.

  ```
  SQL> exec dbms_ra.key_rekey;
  ```

- Re-key specific a protected database.

  ```
  SQL> exec dbms_ra.key_rekey (db_unique_name=>'< DB UNIQUE NAME >');
  ```

- Re-key ~*all*~ protected databases for a specific protection policy.

  ```
  SQL> exec dbms_ra.key_rekey (protection_policy_name=>'< PROTECTION POLICY
  >>');
  ```

Re-keying creates new TDE master keys that are used from that point in time forward. Re-keying does not affect the availability of older master keys in the keystore.

Copyright © 2014, 2026, Oracle and/or its affiliates.

12

Implementing Immutable Backups

Regulatory and cyber security requirements need *"immutability"*, a feature that prevents backup deletion and modification by any user. *"Immutable backups"* are backups that cannot be deleted or modified. Some government regulations have specific rules for *compliance retention* and *legal holds*.

Recovery Appliance Immutable (Compliance) backups adhere to a *recovery window compliance* period that is set in the protection policy. The Recovery Appliance enforces the backup retention to recover to any point-in-time in the compliance period (e.g. 7 days).

In contrast, the *recovery window goal* spans compliance period and beyond (e.g. total of 30 days) for operational backup retention needs. The Recovery Appliance attempts to meet all of its goals using the available space, but does not strictly enforce.

This section describes the different settings available in the Recovery Appliance to indicate which backups are immutable. By using these features, backups are prevented from being prematurely deleted by automated processes, by mistake, or by malicious users. Also, administrators and automated processes are prevented from adjusting downward the `KEEP UNTIL` time.

An immutable backup uses space that cannot be freed until the backup's immutability period ends (`KEEP UNTIL`) or the compliance condition (`COMPLIANCE_HOLD`) is removed. This might prevent new backups from being created on the system.

Compliance retention is an ongoing retention window and protection policy level setting. Legal hold is a temporary suspension of retention and expiration rules on individual database backups.

The `KEEP` and `KEEP UNTIL` attributes are also used with Archival Backups, another option for archive to cloud or tape. An archival backup is a full backup that includes required archived logs to recover the data file backups to a specific point in time and has a defined retention period.

> **Note**
>
> The Oracle Zero Data Loss Recovery Appliance only enforces backup immutability within its domain, within its storage. Recovery Appliance cannot enforce backup immutability on tape or in the cloud, where other services must take on the responsibility for enforcing immutability of backups.

The Oracle Zero Data Loss Recovery Appliance addresses immutable backups through attributes for Enterprise Manager Cloud Control, the APIs, the protection policies, and the jobs for backup to tape and cloud.

Legal Holds

A *"legal hold"* is a process that an organization uses to preserve all forms of potentially relevant information when litigation is pending or reasonably anticipated. When initiated, a legal hold requires that the organization suspend the normal disposition of obsolete records.

Copyright © 2014, 2026, Oracle and/or its affiliates.

The Recovery Appliance administrator can create a legal hold on existing disk backups for specific databases. Backups on legal hold cannot be deleted by internal processes or administrator commands, until the hold is disabled.

This is configured by enabling the `COMPLIANCE_HOLD` attribute with UPDATE_DB for a specified database. A starting date for the hold must be within the current recovery window available for the database. All backups from this date onwards are protected from being deleted. The metadata for those backups is assigned the `COMPLIANCE_HOLD` attribute that prevents the backup from being deleted by automated processes or administrators. Legal hold backups are indefinitely retained until the hold is disabled. A legal hold is meant to be transitive and not permanent for a database.

`COMPLIANCE_HOLD` applies to the storage of the Recovery Appliance. Compliance hold backups on the Recovery Appliance that are archived to cloud or tape are treated as normal archived backups along with deletion of obsolete backups (`recovery_window_sbt`). Therefore, to ensure legal hold on cloud or tape, immutability settings must also be configured using the administrative interfaces for those locations. If a database is in `COMPLIANCE_HOLD` and the Recovery Appliance attempts to delete the backup piece on tape or cloud, tape or cloud location grants or denies the request. If tape or cloud refuses to delete a piece, the pointer to the piece inside of the Recovery Appliance is preserved. In this manner, all cloud and tape backup records are preserved in the Recovery Appliance, because the destination blocks any delete operations issued by the Recovery Appliance.

> **Note**
>
> `COMPLIANCE_HOLD` can prevent the addition of new backups to the Recovery Appliance, when backups associated with the legal hold fill up the storage of the Recovery Appliance, because old backups aren't *"expiring"* and having their storage reclaimed.

Compliance Backup and Reserved Space

Reserved space is the minimum allotted space necessary for each database's backups. This space covers backups needed to meet *recovery window goal*. Depending on backup activity and total available space, the actual space used can be more than reserved space.

For databases set by their protection policy for compliance backups, the reserved space becomes the *maximum space under compliance*. If there is high backup activity and total space needed to meet *recovery window compliance* exceeds reserved space, backups will fail. Reserved space must be increased so that backups continue to be received. In fact, it is a challenge to monitor and adjust the reserved space for hundreds and thousands of databases that grow and shrink over time.

Auto-Tuned Reserved Space is a feature available for all databases, and is beneficial for compliance backups. It automatically manages the reserved space for compliance backups and reduces the administrator involvement in space management. The reserved space is automatically adjusted when a backup is received that exceeds reserved space setting for that database.

12.1 Managing Recovery Window Compliance

The *"Recovery Window Compliance"* is a range of time that the Recovery Appliance will ensure databases can be recovered from their backups. This is specified with a `RECOVERY_WINDOW_COMPLIANCE` attribute in the protection policy. When set in the protection

Copyright © 2014, 2026, Oracle and/or its affiliates.

policy, newly created backups of that policy are held on the Recovery Appliance for that period of time.

`RECOVERY_WINDOW_COMPLIANCE` is different and more restrictive than `RECOVERY_WINDOW_GOAL`, because the *goal* doesn't have to be met but the *compliance* does. The goal might be for the Recovery Appliance to recover a given database to any point in the last 30 days, if reserve storage is sufficient and not needed and overwritten by newer backups. Recovery window compliance might require the Recovery Appliance to recover a given database to any point in the past 7 days regardless of reserve storage constraints.

> **Note**
>
> If the `RECOVERY_WINDOW_COMPLIANCE` is too large, it can prevent the addition of new backups to the Recovery Appliance, because reserve storage isn't available. When `RECOVERY_WINDOW_COMPLIANCE` consumption is near the reserved storage limit and an incoming backup piece would have the space used exceed that limit, RMAN fails immediately.

Because backups need space, you must estimate how much reserve space you believe is needed to store backups. The ESTIMATE_SPACE procedure can assist with determining reserved space. The `target_window` used to estimate space should be the `RECOVERY_WINDOW_COMPLIANCE` ***plus an extra day*** for edge conditions.

Changes can be made to the protection policy to keep backups longer or shorter for new backups. However, once `RECOVERY_WINDOW_COMPLIANCE` is set for a given backup, it is strictly enforced and the backup is not deleted until the `RECOVERY_WINDOW_COMPLIANCE` period expires.

The two main methods for creating and maintaining recovery window compliance are with an application, such as **Enterprise Manager Cloud Control**, or using the **DBMS_RA** API. In either case, the

- CloudControl
- API

CloudControl

The steps for setting and removing compliance hold on a database using Enterprise Manager Cloud Control are:

1. Log in to your Cloud Control page.

   > **See Also**
   >
   > "Accessing the Recovery Appliance Home Page" for more information.

2. From any Cloud Control page, use the **Targets** drop-down menu and select **Recovery Appliances**.

   The Recovery Appliances page appears.

3. In the **Name** column, click the name of a Recovery Appliance.

Copyright © 2014, 2026, Oracle and/or its affiliates.

The **Home** page for the selected Recovery Appliance appears.

4. From the **Recovery Appliances** drop-down menu, select **Protection Policies**.

   This displays a table with all of the protection policies that the Recovery Appliance is currently enforcing.

5. Select the **Protected Policy** table and then on **Edit** to make change for its recovery window compliance or to turn on keep compliance.

   This opens the **Update Protection Policy** dialog box.

> **Note**
>
> Changes to the protection policy affect all databases that use that policy, which are listed below the **Protected Policy** table for the policy selected.

The **Recovery Window Compliance** is a time range to which the Recovery Appliance must ensure that all databases using that protection policy can be recovered. This should be smaller than **Recovery Window Goal**. The Recovery Window Compliance may be null. If too large, this can result in the Recovery Appliance rejecting new backups, because old backups for compliance purposes have not *"expired"* and made their storage space available for re-use with incoming backups.

The protection policy can also be used to establish **Keep Compliance**. When enabled in the protection policy, the Recovery Appliance keeps the backups of all of the associated databases until the *"keep until time"*.

Later when a protection policy and its associated databases no longer require a compliance hold, be sure to remove.

API

PL/SQL Snippets on Setting Immutability Settings in Protection Policies

The protection policy has two new immutability settings and `UPDATE_DB` has one.

If you are creating a new protection policy for compliance, refer to Creating a Protection Policy. You can set multiple compliance attributes at the same time, such as in the following snippet.

```
dbms_ra.CREATE_PROTECTION_POLICY (
PROTECTION_POLICY_NAME => ‘Policy 1’,
STORAGE_LOCATION_NAME => ‘DELTA’,
RECOVERY_WINDOW_GOAL = INTERVAL '14' DAY,
RECOVERY_WINDOW_COMPLIANCE => INTERVAL '7' DAY,
KEEP_COMPLIANCE => ‘YES’,
ALLOW_BACKUP_DELETION => ‘NO’);
```

If you are modifying existing protection policies for compliance rules, here are PL/SQL snippets on updating a policy.

- **Set `RECOVERY_WINDOW_COMPLIANCE` settings for one or more protection policies.**

```
BEGIN
DBMS_RA.UPDATE_PROTECTION_POLICY(
    PROTECTION_POLICY_NAME =>  '&pname',
    RECOVERY_WINDOW_GOAL => INTERVAL '92' DAY,
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
    RECOVERY_WINDOW_COMPLIANCE => INTERVAL '14' DAY);
END;
```

> **Note**
>
> Exercise caution in setting this value, because too large values for `RECOVERY_WINDOW_COMPLIANCE` can result in the Recovery Appliance running out of storage with backups that can't be deleted (yet) to the point where new backups can't be stored and are rejected. This number for `RECOVERY_WINDOW_COMPLIANCE` should be less than the `RECOVERY_WINDOW_GOAL`. `RESERVED_SPACE` needs to be large enough to support all needed backups for compliance retention, otherwise the space could fill and cause new backups to be rejected.

- **Set `ALLOW_BACKUP_DELETION` attribute to `NO` for one or more protection policies.**

```
BEGIN
DBMS_RA.UPDATE_PROTECTION_POLICY(
    PROTECTION_POLICY_NAME =>  '&pname',
    ALLOW_BACKUP_DELETION =>  'NO');
END;
```

`ALLOW_BACKUP_DELETION` set to `NO` means that the Recovery Appliance does not allow deletion of these backups, which is the requirement of a legal hold. .

`ALLOW_BACKUP_DELETION` set to `YES` means that the Recovery Appliance allows deletion of these backups when they expire beyond their recovery window goals.

> **Note**
>
> `ALLOW_BACKUP_DELETION` has to be set to `NO` (disabled) before `KEEP_COMPLIANCE` is enabled.

- **Enable `KEEP_COMPLIANCE` immutable settings for one or more protection policies.**

  Here is a pseudo snippet for PL/SQL that shows the `KEEP_COMPLIANCE` attribute being set in a given protection policy.

```
BEGIN
DBMS_RA.UPDATE_PROTECTION_POLICY(
  PROTECTION_POLICY_NAME =>  '&pname',
  KEEP_COMPLIANCE => 'YES');
END;
```

`YES`: The Recovery Appliance prevents the deletion of `KEEP` backups. `NO`: The administrator of the Recovery Appliance is permitted to remove `KEEP` backups.

The `KEEP_COMPLIANCE` attribute helps enable the archival backup by preventing its storage from getting overwritten when the backup would normally have expired according to its recovery window goals. However, once the `keep_time` is reached, the backup can be deleted.

Copyright © 2014, 2026, Oracle and/or its affiliates.

12.2 Managing Compliance Holds

A *"compliance hold"* or *"legal hold"* is a process that an organization uses to preserve all forms of potentially relevant information when litigation is pending or reasonably anticipated. When initiated, a compliance hold requires that the organization suspend the normal disposition of obsolete records.

The Recovery Appliance administrator can create a compliance hold on existing disk backups for specific databases. Backups on compliance hold cannot be deleted by internal processes or administrator commands, until the compliance hold is disabled.

`COMPLIANCE_HOLD` applies to the storage of the Recovery Appliance. Compliance hold backups on the Recovery Appliance that are archived to cloud or tape are treated as normal archived backups along with deletion of obsolete backups (`recovery_window_sbt`). Therefore, to ensure legal hold on cloud or tape, immutability settings must also be configured using the administrative interfaces for those locations. If a database is in `COMPLIANCE_HOLD` and the Recovery Appliance attempts to delete the backup piece on tape or cloud, tape or cloud location grants or denies the request. If tape or cloud refuses to delete a piece, the pointer to the piece inside of the Recovery Appliance is preserved. In this manner, all cloud and tape backup records are preserved in the Recovery Appliance, because the destination blocks any delete operations issued by the Recovery Appliance.

> **Note**
>
> `COMPLIANCE_HOLD` can prevent the addition of new backups to the Recovery Appliance, when backups associated with the legal hold fill up the storage of the Recovery Appliance, because old backups aren't *"expiring"* and having their storage reclaimed.

The two main methods for creating and maintaining compliance holds are with an application, such as **Enterprise Manager Cloud Control**, or using the **DBMS_RA** API.

- CloudControl
- API

CloudControl

The steps for setting and removing compliance hold on a database using Enterprise Manager Cloud Control are:

1. Log in to your Cloud Control page.

   > **See Also**
   >
   > "Accessing the Recovery Appliance Home Page" for more information.

2. From any Cloud Control page, use the **Targets** drop-down menu and select **Recovery Appliances**.

Copyright © 2014, 2026, Oracle and/or its affiliates.

The Recovery Appliances page appears.

3. In the **Name** column, click the name of a Recovery Appliance.

   The **Home** page for the selected Recovery Appliance page appears.

   From this page you can see a snapshot of the entire Recovery Appliance, and also click links to obtain more information about a particular area.

4. From the **Recovery Appliances** drop-down menu, select **Protected Databases**.

   This displays a table with all of the databases that the Recovery Appliance is currently protecting.

5. Select the row in the **Protected Databases** table for the database that needs compliance to be turned on. While highlighted, select the **Set Compliance** button from above the table.

   This opens the **Set Compliance Hold** dialog box with the checkbox to set or remove a compliance hold on that database.

6. To set a compliance hold, specify a start date that is within the current recovery window for the database and mark the checkbox **Compliance Hold**.

   All backups from the specified date onwards will not be deleted.

Later when a given database no longer requires a compliance hold, be sure to remove.

API

A *"compliance hold"* is configured by enabling the `COMPLIANCE_HOLD` attribute with UPDATE_DB for a specified database. A starting date for the hold must be within the current recovery window available for the database. All backups from this date onwards are protected from being deleted. The metadata for those backups is assigned the `COMPLIANCE_HOLD` attribute that prevents the backup from being deleted by automated processes or administrators. Legal hold backups are indefinitely retained until the hold is disabled. A legal hold is meant to be transitive and not permanent for a database.

PL/SQL Snippets on Setting Immutability Settings in Protection Policies

The protection policy has two new immutability settings and `UPDATE_DB` has one.

If you are creating a new protection policy for compliance, refer to Creating a Protection Policy. You can set multiple compliance attributes at the same time, such as in the following snippet.

```
dbms_ra.CREATE_PROTECTION_POLICY (
PROTECTION_POLICY_NAME => ‘Policy 1’,
STORAGE_LOCATION_NAME => ‘DELTA’,
RECOVERY_WINDOW_GOAL = INTERVAL '14' DAY,
RECOVERY_WINDOW_COMPLIANCE => INTERVAL '7' DAY,
KEEP_COMPLIANCE => ‘YES’,
ALLOW_BACKUP_DELETION => ‘NO’);
```

If you are modifying existing protection policies for compliance rules, here are PL/SQL snippets on updating a policy.

- **Set `COMPLIANCE_HOLD` settings for one or more databases.**

  ```
  BEGIN
  DBMS_RA.UPDATE_DB(
      DB_UNIQUE_NAME =>  '&dbname',
  ```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
    COMPLIANCE_HOLD => SYSTIMESTAMP - NUMTODSINTERVAL(7, 'DAY');
END;
```

`COMPLIANCE_HOLD` is the time from which backups may not be deleted from the Recovery Appliance. The database must be recoverable starting at the time specified by this `COMPLIANCE_HOLD`. Specify the time as any valid `TIMESTAMP WITH TIME ZONE` expression.

If an immutable cloud location is configured (via OCI Console) with an indefinite retention policy on the bucket, the `COMPLIANCE_HOLD` attribute on the database also prevents deletion of backups from the hold period that were archived to the cloud location, until the `COMPLIANCE_HOLD` is removed.

- **Set `ALLOW_BACKUP_DELETION` attribute to `NO` for one or more protection policies.**

```
BEGIN
DBMS_RA.UPDATE_PROTECTION_POLICY(
    PROTECTION_POLICY_NAME =>  '&pname',
    ALLOW_BACKUP_DELETION =>  'NO');
END;
```

`ALLOW_BACKUP_DELETION` set to `NO` means that the Recovery Appliance does not allow deletion of these backups, which is the requirement of a legal hold. .

`ALLOW_BACKUP_DELETION` set to `YES` means that the Recovery Appliance allows deletion of these backups when they expire beyond their recovery window goals.

> **Note**
>
> `ALLOW_BACKUP_DELETION` has to be set to `NO` (disabled) before `KEEP_COMPLIANCE` is enabled.

- **Enable `KEEP_COMPLIANCE` immutable settings for one or more protection policies.**

  Here is a pseudo snippet for PL/SQL that shows the `KEEP_COMPLIANCE` attribute being set in a given protection policy.

```
BEGIN
DBMS_RA.UPDATE_PROTECTION_POLICY(
  PROTECTION_POLICY_NAME =>  '&pname',
  KEEP_COMPLIANCE => 'YES');
END;
```

`YES`: The Recovery Appliance prevents the deletion of `KEEP` backups. `NO`: The administrator of the Recovery Appliance is permitted to remove `KEEP` backups.

The `KEEP_COMPLIANCE` attribute helps enable the archival backup by preventing its storage from getting overwritten when the backup would normally have expired according to its recovery window goals. However, once the `keep_time` is reached, the backup can be deleted.

Copyright © 2014, 2026, Oracle and/or its affiliates.

13

Archival Backups

The Archival Backup is a full backup with `KEEP` specified for a specific point in time and intended to be held in tape or cloud storage until a future point in time.

An archival backup is used to maintain a fixed recovery point on cloud or tape. It has advantages over the present operational strategy to satisfy any point-in-time recovery within the `recovery_window_sbt` period, because the archival backup doesn't have to regularly copy incremental backups and archived log backups.

For example, an organization has a security requirement to generate a monthly full backup and to keep five years of these monthly backups available in cloud storage. To achieve this, the organization schedules archival backup command to be executed at the end of each month with `KEEP UNTIL SYSDATE+5 YEARS`.

The Recovery Appliance supports archival backups sent to tape or cloud.

You can convert existing tape configurations to cloud configurations using the `racli convert tape_to_cloud` command. See Converting Tape Configurations to Cloud for details.

> **Note**
>
> The databases must have archived log mode turned on. The CREATE_ARCHIVAL_BACKUP command requires archive logs to properly compute the necessary files to create a complete consistent backup for archival purposes. Databases with archive log mode off must continue to make KEEP backups of the protected database and then use the MOVE_BACKUP command to archive onto tertiary storage.
>
> When the backups are created as *space-efficient backups* (using compression and TDE encryption), their restoration requires the Recovery Appliance, because the database communicates with the appliance to process space-efficient backups. As such, these backups cannot be restored directly from the media.

13.1 Managing Archival Backups

The two main methods for creating and maintaining archival backups are with an application, such as **Enterprise Manager Cloud Control**, or using the **DBMS_RA** API.

- CloudControl
- API

CloudControl

The steps for accessing and creating archival backups using Enterprise Manager Cloud Control are:

Copyright © 2014, 2026, Oracle and/or its affiliates.

1. Log in to your Cloud Control page.

> **See Also**
>
> "Accessing the Recovery Appliance Home Page" for more information.

2. From any Cloud Control page, use the **Targets** drop-down menu and select **Recovery Appliances**.

   The Recovery Appliances page appears.

3. In the **Name** column, click the name of a Recovery Appliance.

   The **Home** page for the selected Recovery Appliance page appears.

   From this page you can see a snapshot of the entire Recovery Appliance, and also click links to obtain more information about a particular area.

4. A **Media Manager** needs to exist or be created for the archival backup, because it defines parameters that are passed to media management software (e.g., Oracle Secure Backup) when backups are copied to media by the Recovery Appliance.

   From the **Recovery Appliances** drop-down menu, select **Media Manager**.

   The **Media Manager** page has the **Media Manager Libraries** table and **Attribute Sets** table. When creating or editing a **Media Manager**, you specify **Name** of the library, **Maximum Channels** that media library has access to, **Restore Channels** reserved for restore operations, **Media Management Vendor Parameters**, and **Media Management Vendor Commands**.

5. Verify that an appropriate **Media Manager** exists for your archival backup to use.

6. From the **Recovery Appliances** drop-down menu, select **Protected Databases**.

   This displays a table with all of the databases that the Recovery Appliance is currently protecting.

7. Select the row in the **Protected Databases** table for the database that needs an archival backup. While highlighted, select the **Archival Backups** button from above the table.

   This opens the **Archival Backups** page that shows a table for the backups with **Restore Point Name**, **Status**, **SCN**, **Restore Tag**, **Rention Time**, and **Created** information.

8. On the **Archival Backups**, click on the **Create Archival Backup** button.

   This opens the **Create Archival Backup** dialog with the options:

   - **Backup on a recurring schedule**: on a specific day and time in one or more months.
   - **One-time archival Backup**: for **Point in Time**, **SCN**, and **Restore Point** types of archival backups, and fields for **Point in Time** and **Restore Point Name**.

   For either option, the **Retention Time** section provides drop-down controls for other important settings on the archive such as **Keep For** and time period; and **Properties** on the archive such as **Attribute Set** (the specific **Media Manager**), **Format**, **Encryption Algorithm**, and **Compression Algorithm**.

> **Note**
>
> The tape or cloud destination for the archival backup are specified through the **Media Manager**.

Copyright © 2014, 2026, Oracle and/or its affiliates.

API

The CREATE_ARCHIVAL_BACKUP procedure creates the archival backup.

Archival backups are controlled by the KEEP_COMPLIANCE attribute in the protection policy: This attribute (when YES) prevents KEEP backups from having their KEEP UNTIL time adjusted down by a database administrator. (Reducing the KEEP UNTIL time is non-compliant and is a method to delete a backup by expiring it early and then its storage space can be reclaimed.) With KEEP_COMPLIANCE, the backups remain available in storage until the specified date, and only then is their storage space reclaimed.

In the following PL/SQL snippet, a single archival backup is created for the database DB_UNIQUE_NAME with a specific restoration period and an expiration date (KEEP_UNTIL_TIME).

```
DBMS_RA.CREATE_ARCHIVAL_BACKUP(
db_unique_name => DB_UNIQUE_NAME,
from_tag => NULL,
compression_algorithm => 'LOW',
encryption_algorithm => NULL,
restore_point => NULL,
restore_until_scn => NULL,
restore_until_time => TO_TIMESTAMP(last_day(sysdate-1)||' 11:59:59 PM'),
attribute_set_name => 'SEVEN_YEAR_VAULT_DRIVE',
format => NULL,
autobackup_prefix => NULL,
restore_tag => NULL,
keep_until_time => TO_TIMESTAMP(ADD_MONTHS(last_day(sysdate-1), 84)||'
11:59:59 PM'),
comments => NULL
max_redo_to_apply => 21);
```

The snippet above can become the body of a loop. In the following snippet for creating archival backups, the two SELECT instances identify databases with specific protection policies, and then filters for databases having restoration backups to archive. Each database R1.DB_UNIQUE_NAME from the selection set has an archival backup created.

```
BEGIN
FOR R1 IN
(
WITH BACKUP_RANGE AS
(select
B.DB_UNIQUE_NAME,B.NZDL_ACTIVE,A.DB_KEY,B.policy_name,A.LOW_TIME,A.HIGH_TIME,
CASE WHEN TO_DATE('2022-10-31','YYYY-MM-DD') between low_time and high_time
THEN 'PASSED' ELSE 'NO_RANGE' END as DISK_RANGE_STATUS,
to_char((SYSTIMESTAMP AT TIME ZONE TO_CHAR(B.timezone)) -
B.minimum_recovery_needed, 'DD-MON-YY HH:MI:SS AM')  LAST_BACKUP_TIME
from
RA_DISK_RESTORE_RANGE A, ra_database B
WHERE b.db_key = a.db_key
AND (B.policy_name like 'GOLD' or B.policy_name like 'SILVER')
ORDER BY B.DB_UNIQUE_NAME,HIGH_TIME DESC)
SELECT
DB_UNIQUE_NAME,NZDL_ACTIVE,DB_KEY,POLICY_NAME,LOW_TIME,HIGH_TIME,DISK_RANGE_ST
ATUS,LAST_BACKUP_TIME
from BACKUP_RANGE
WHERE DISK_RANGE_STATUS='PASSED' AND NZDL_ACTIVE='YES'
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
ORDER BY DB_UNIQUE_NAME
)
LOOP
dbms_output.put_line('Submitting Archival Backup for:'||
R1.DB_UNIQUE_NAME||'LAST_BACKUP_TIME:='||R1.LAST_BACKUP_TIME);

DBMS_RA.CREATE_ARCHIVAL_BACKUP(
db_unique_name => R1.DB_UNIQUE_NAME,
from_tag => NULL,
compression_algorithm => 'LOW',
encryption_algorithm => NULL,
restore_point => NULL,
restore_until_scn => NULL,
restore_until_time => TO_TIMESTAMP(last_day(sysdate-1)||' 11:59:59 PM'),
attribute_set_name => 'SEVEN_YEAR_VAULT_DRIVE',
format => NULL,
autobackup_prefix => NULL,
restore_tag => NULL,
keep_until_time => TO_TIMESTAMP(ADD_MONTHS(last_day(sysdate-1), 84)||'
11:59:59 PM'),
comments => NULL
max_redo_to_apply => 21);

dbms_output.put_line('========================================');
END LOOP;
END;
/
```

Table 13-1 DBMS_RA Procedures Associated with Tape/Cloud/Archive Backup Operations

SBT Object	Procedures
SBT Job	• CREATE_SBT_JOB_TEMPLATE • UPDATE_SBT_JOB_TEMPLATE • DELETE_SBT_JOB_TEMPLATE
SBT Library	• CREATE_SBT_LIBRARY • UPDATE_SBT_LIBRARY • PAUSE_SBT_LIBRARY • RESUME_SBT_LIBRARY • DELETE_SBT_LIBRARY
SBT Attribute Set	• CREATE_SBT_ATTRIBUTE_SET • UPDATE_SBT_ATTRIBUTE_SET • DELETE_SBT_ATTRIBUTE_SET
Protection Policy	• CREATE_PROTECTION_POLICY • UPDATE_PROTECTION_POLICY • DELETE_PROTECTION_POLICY
Backup	• QUEUE_SBT_BACKUP_TASK • COPY_BACKUP • MOVE_BACKUP

Copyright © 2014, 2026, Oracle and/or its affiliates.

Copyright © 2014, 2026, Oracle and/or its affiliates.

14

Recovery Appliance Partners

To assist and simplify remote management of Recovery Appliances in replication and migration scenarios, Recovery Appliance partnerships, or *RA partners*, are established. This limited partnership establishes a two-way connection between a source and destination Recovery Appliance that allows information, such as status, to be retrieved from a remote Recovery Appliance. The partnership also allows altering of objects on the remote Recovery Appliance, such as changing the TLS configuration of the downstream Recovery Appliance to match the upstream.

The RA partnership is established using racli add ra_partner. It automates common administrative tasks across partners, which reduces the potential for human error and the need to remotely log into each system. It creates a non-root `'partner_user'` with limited privileges, and sets up SSH key-based two-way authentication.

Copyright © 2014, 2026, Oracle and/or its affiliates.

15

Replicating Backups with Recovery Appliance

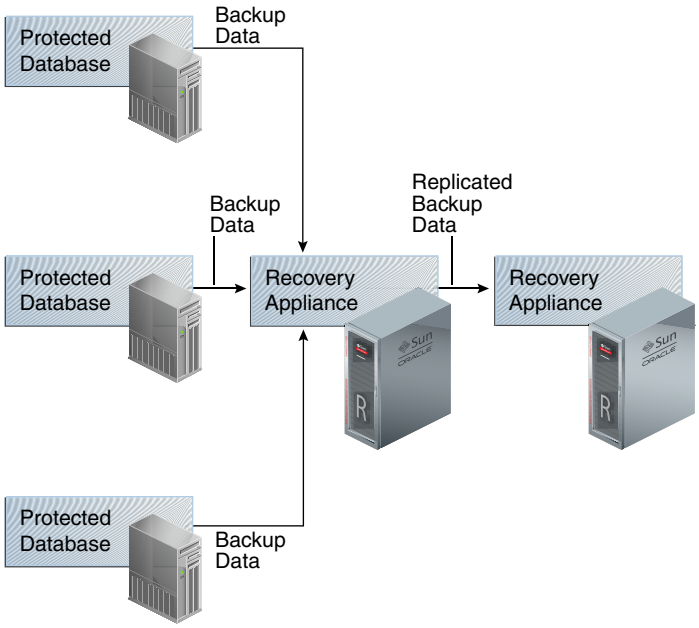
This chapter explains replication of Recovery Appliances for disaster recovery. It provides examples of replication topologies and concludes with three different methods for configuring replication.

- Cloud Control, or Enterprise Management: This solution might be appropriate for replication if your environment already uses such tools.
- RACLI Commands Relating to Replication: This command line solution is appropriate for most replication situations, because these commands are wrappers for several underlying steps to make configuration and management easier for administrators. Being in the RACLI environment, more control exists over user privileges and in logging actions.
- DBMS_RA Procedures Relating to Replication: The `DBMS_RA` API solution is appropriate for environments that have custom programming and automation. This allows tighter integration into such systems. The `DBMS_RA` is more cumbersome but reliable, because it came first and is the API called by wrapper functions in Cloud Control and RACLI.

15.1 Overview of Recovery Appliance Replication

In the example Figure 15-1, a protected database sends backups to one Recovery Appliance, which transfers the backups on to another Recovery Appliance. This topology is called one-way Recovery Appliance replication. The first Recovery Appliance is the upstream Recovery Appliance and the second is the downstream Recovery Appliance.

Figure 15-1 Simple Replication Topology



Copyright © 2014, 2026, Oracle and/or its affiliates.

Many other replication scenarios are possible, as given in Additonal Replication Topology Examples

15.1.1 Protection Policies for Replication

Replication for a protected database occurs when the following conditions are met:

- On the upstream Recovery Appliance, a replication server configuration specifies a Recovery Appliance acting as a downstream replication Recovery Appliance (`CREATE_REPLICATION_SERVER`).
- On the upstream Recovery Appliance, a protection policy is associated with the replication server configuration (`ADD_REPLICATION_SERVER`).
- On the upstream Recovery Appliance, a protected database is assigned (`ADD_DB`) to the protection policy associated with the replication server configuration.
- On the downstream Recovery Appliance, creating the protection policy (`CREATE_PROTECTION_POLICY`), adding the databases to the policy (`ADD_DB`), and granting replication acess to the database (`GRANT_DB`) should be done first in the overall setup.

By default, after performing `ADD_REPLICATION_SERVER`, the latest full backup is replicated for each database in the policy. If `COPYALL_BACKUP` is set to `TRUE`, all backups (fulls, incrementals, archived logs) are replicated. This replication mode can be used as part of migrating backups from one appliance to another. Thereafter, the upstream Recovery Appliance replicates new backups as it receives them.

15.1.2 How Recovery Appliance Replicates Backups: Basic Process

Assume that a protected database backs up to a Recovery Appliance using the incremental-forever policy. When an protected database sends a backup to a Recovery Appliance configured for replication, the following basic steps occur:

1. The upstream Recovery Appliance ingests the backup, checking the protection policy to determine whether it is associated with a replication server configuration.
2. If a replication server configuration exists for the protection policy, then the upstream Recovery Appliance replicates the backup. The replication process includes:
   - Creating metadata records to track the replicated records

     Note

     When real-time redo transport is enabled, incoming redo changes are not replicated in real time by Recovery Appliance. When an archived redo log backup is created, the Recovery Appliance automatically replicates this backup along with the data file backups.

   - Transferring the data blocks over the network to each specified downstream Recovery Appliance
3. The downstream Recovery Appliance ingests the backup, creating a virtual backup.

Copyright © 2014, 2026, Oracle and/or its affiliates.

> **Note**
>
> The ingest phase on the downstream is the same as the ingest phase described in Step 1. Thus, if the *downstream* Recovery Appliance is also configured to replicate the backup, then it assumes the role of an *upstream* Recovery Appliance, and then replicates the backup to the Recovery Appliances that are directly downstream, and so on.

4. Shortly afterward, the upstream Recovery Appliance sends a reconcile request to the downstream Recovery Appliance, which in turn sends metadata about the backup to the upstream Recovery Appliance.

   In Recovery Appliance replication, reconciling is the process by which a Recovery Appliance receives metadata from the Recovery Appliances that are immediately downstream.

Thus, after the backup is replicated, both the upstream and downstream recovery catalog have a record of the protected database backup.

15.1.3 Archived Backup Mirroring

Archived backups mirroring (SBT mirroring) allows a downstream Recovery Appliance configured as a replication partner to access externally stored backups on tape, cloud or third party storage that were created by an upstream Recovery Appliance.

The local SBT library is configured by RACLI, or a system administrator, to access the same external storage, even if the external storage wasn't orginally set up with the remote Recovery Appliance.

Prior to RA 23.1, a downstream Recovery Appliance has a copy of the backups and catalog records for the upstream Recovery Appliance, but has no information or access to the archived backups on external storage. Using archived backups mirroring starting in RA 23.1, the remote Recovery Appliance has details about this backup from the source replicated Recovery Appliance.

Copy-to-tape and archived-to-cloud backup records are mirrored between the Recovery Appliance catalogs in a replication pair, so that either Recovery Appliance can access backups from tape/cloud that are needed for restore operations. This is enabled by setting `sbt_mirroring=>'TRUE'` in the SBT library configured for tape/cloud with the same SBT library name on each Recovery Appliance.

> **Note**
>
> The user must assure that both Recovery Appliances point to the same external storage before enabling.

The local SBT library is linked with the replicated backup piece using the SBT library mapping. This allows externally stored replication backup-pieces to be locally available for restore operations.

CREATE_SBT_LIBRARY and UPDATE_SBT_LIBRARY help identify the library metadata (`SBT_LIB_DESC`) with a new column `sbt_mirror`. If the library has been marked to be mirrored but there is no corresponding partner on the other Recovery Appliance then, there won't be any library information exchange during reconcile and everything proceeds as before.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Requirements

- The SBT library be setup on a remote Recovery Appliance such that the remote SBT library has the same name and accesses the same external storage destination as its partner Recovery Appliance.
- The same library name is used across the replicating pair of Recovery Appliance. The library name on a Recovery Appliance is always unique, so this uniqueness along with library type and replication pair details help map the SBT libraries across Recovery Appliances. The library type can be "T" (tape) or "C" (cloud).
- The SBT mirror can be setup without automation with the API. For migration scenarios between Recovery Appliance's, RACLI migration automates the setup of all configurations needed on the destination Recovery Appliance, including the SBT libraries.
- The automation layer is also required to check whether there are any library name conflicts on paired Recovery Appliance before doing the setup

Using SBT Mirroring

All the metadata mapping between the partner Recovery Appliances is done via `reconcile`. This synchronizes the catalog metadata between the replicating parnter Recovery Appliances and allows the upstream Recovery Appliance to access the downstream backup metadata.

After setting up mirroring, it won't take effect instantly. A reconcile and/or next backup, which would ultimately initiate an implicit reconcile, make the changes.

1. User calls CREATE_SBT_LIBRARY or UPDATE_SBT_LIBRARY with `sbt_mirror` set to `YES`. This calls the required setup automation.
2. The triggered automation looks at the associated replicating Recovery Appliance and initiates a SBT library creation with library name and type specifications mentioned above.
3. On the paired Recovery Appliance, the automation makes sure there does not already exist any SBT library with same name.
4. The automation layer creates a new SBT library (with `sbt_mirror` set to `YES`) on the partner Recovery Appliance that can access the same external storage location as the Recovery Appliance initiating this automation (keeping the library name same).

15.1.4 How RMAN Restores Backups in a Replication Environment

To restore a protected database, RMAN typically connects `AS CATALOG` to the same Recovery Appliance to which it originally sent backups. For example, in Figure 15-2, if RMAN needed to restore `orcl11`, then RMAN would connect to the catalog on the upstream Recovery Appliance.

If backups exist on any Recovery Appliances in the replication scheme, then the upstream Recovery Appliance can retrieve and restore the backups from the other Recovery Appliances. For example, in Figure 15-2, if RMAN needed to restore `orcl11`, if the backups from the upstream had been purged, and if the backups were still available on the downstream Recovery Appliance, then those backups are transferred to the upstream Recovery Appliance and then to the database.

If necessary, RMAN can also restore a backup directly from a downstream Recovery Appliance. RMAN connects `AS CATALOG` to the downstream Recovery Appliance, and then restores the backup. For example, in Figure 15-2, if RMAN needed to restore `prod3`, but the upstream Recovery Appliance was temporarily inaccessible, then RMAN could connect directly

Copyright © 2014, 2026, Oracle and/or its affiliates.

to the catalog on the downstream Recovery Appliance, and then restore the backups directly to the protected database host.

> **Note**
>
> When using either Oracle Enterprise Manager Cloud Control (Cloud Control) or the command line, restoring backups from a downstream Recovery Appliance requires additional configuration. See *Zero Data Loss Recovery Appliance Protected Database Configuration Guide*.

15.2 Additonal Replication Topology Examples

Replication topologies can be as complex as required. The primary variables are as follows:

- The total number of Recovery Appliances in the replication environment, and their relationships to one another
- The protection policies (`CREATE_PROTECTION_POLICY`) on the upstream Recovery Appliance that manage the *outgoing* replicated backups, and the policies on the downstream Recovery Appliance that manage the *incoming* replicated backups
- The replication server configurations (`CREATE_REPLICATION_SERVER`) that exist on each Recovery Appliance in the replication environment
- The association between a replication server configuration and a protection policy (`ADD_REPLICATION_SERVER`)

You can chain replication so that an upstream Recovery Appliance replicates to multiple downstream Recovery Appliances, while a downstream Recovery Appliance receives backups from multiple upstream Recovery Appliances. A downstream Recovery Appliance can receive both backups from both its own protected databases and replicated backups. Any Recovery Appliance in the replication topology can simultaneously perform the upstream and downstream replication roles.

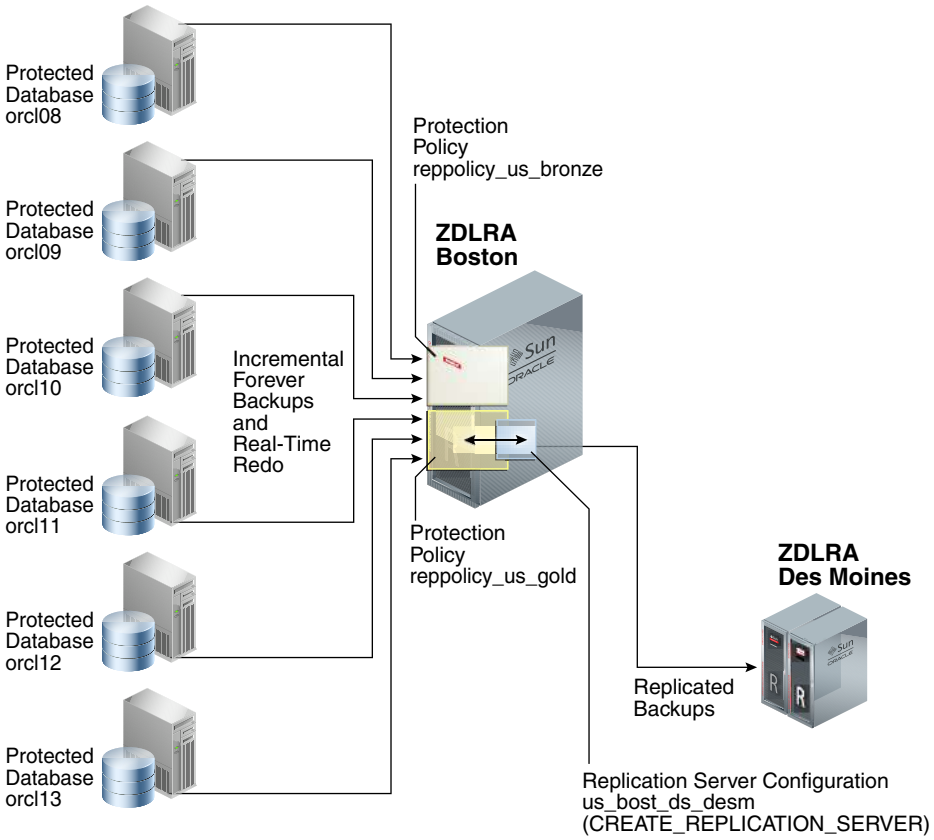
> **Note**
>
> If a Recovery Appliance is both upstream and downstream, then you must configure it for both roles.

15.2.1 Replication to One Downstream Recovery Appliance

Figure 15-2 shows three databases associated with the `reppolicy_us_bronze` protection policy (`orcl08`, `orcl09`, and `orcl10`), and three databases associated with the `reppolicy_us_gold` protection policy (`orcl11`, `orcl12`, and `orcl13`). Only `reppolicy_us_gold` is associated with a replication server configuration, which is named `us_bost_ds_desm`. In this topology, the upstream `ZDLRA Boston` only transfers backups from databases protected by the `reppolicy_us_gold` policy to the downstream `ZDLRA Des Moines`.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Figure 15-2 Databases Replicating to One Recovery Appliance

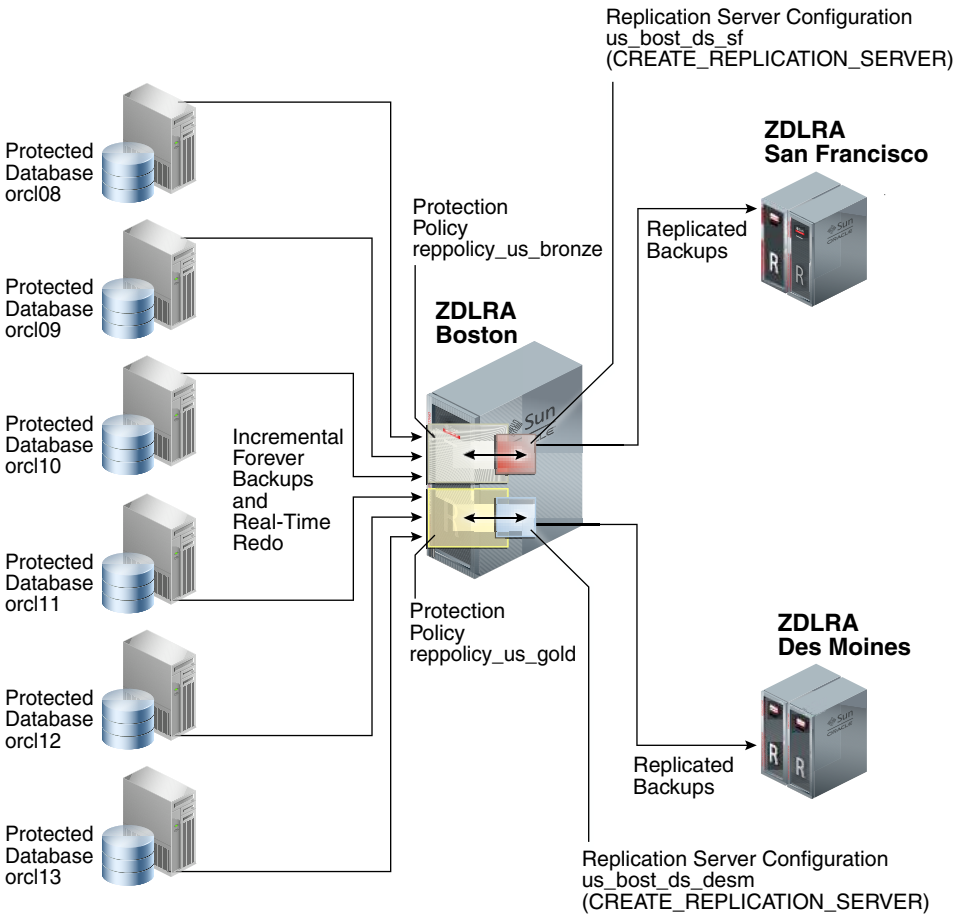


15.2.2 Replication to Multiple Downstream Recovery Appliances

Because each protected database has its own protection policy, each policy can be associated with a different replication server configuration. For example, in Figure 15-3, the `reppolicy_us_bronze` policy is associated with replication server configuration `us_bost_ds_sf`, which replicates backups for databases protected by `reppolicy_us_bronze` (`orcl08`, `orcl09`, and `orcl10`) to the downstream Recovery Appliance named `ZDLRA San Francisco`. The `reppolicy_us_gold` policy is associated with replication server configuration `us_bost_ds_desm`, which replicates backups for databases protected by `reppolicy_us_gold` (`orcl11`, `orcl12`, and `orcl13`) to the downstream Recovery Appliance named `ZDLRA Des Moines`.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Figure 15-3 Databases Replicated to Multiple Recovery Appliances



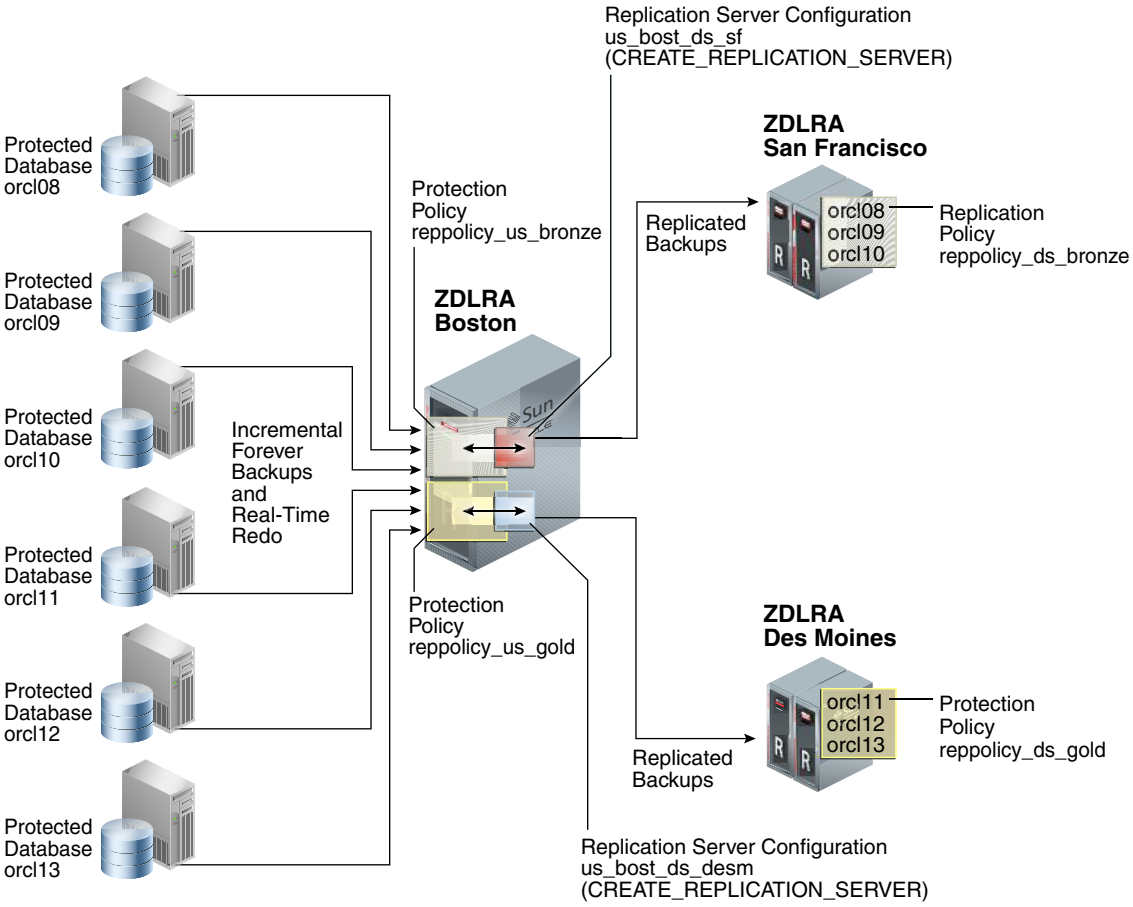
15.2.3 Replication Using Different Policies on Downstream Recovery Appliances

At each downstream Recovery Appliance in a replication scheme, the protection policy defines a disk recovery window goal and tape retention policy for received backups. The downstream configuration is completely independent of the upstream configuration. Thus, you have the flexibility to configure a downstream Recovery Appliance with more storage and longer recovery windows than its upstream Recovery Appliances, for example, using the downstream Recovery Appliance as a longer-term retention backup repository.

Figure 15-4 shows that `reppolicy_us_bronze` backups up on the upstream Recovery Appliance are protected by the `reppolicy_ds_bronze` policy on `ZDLRA San Francisco`. The `reppolicy_us_gold` backups up on the upstream Recovery Appliance are protected by the `reppolicy_ds_gold` policy on `ZDLRA Des Moines`.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Figure 15-4 Different Protection Policies on Each Recovery Appliance

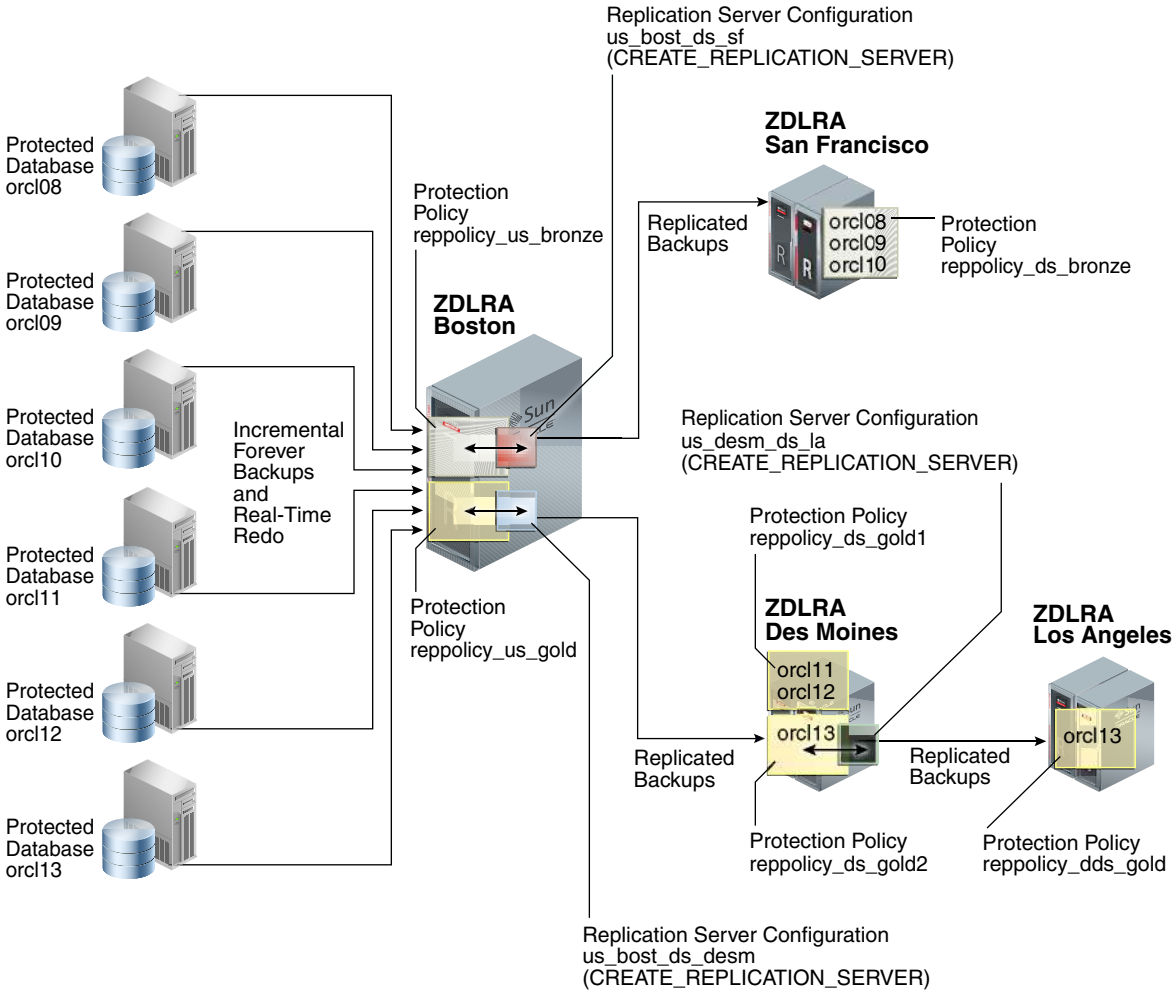


15.2.4 Cascaded Replication

Figure 15-5 shows a more complicated topology. Databases `orcl11`, `orcl12`, and `orcl13` are protected by the `reppolicy_us_gold` protection policy on `ZDLRA Boston`, which is furthest upstream. The `reppolicy_us_gold` policy replicates the backups for these databases to `ZDLRA Des Moines`, which is immediately downstream. However, two separate protection policies exist on `ZDLRA Des Moines`: `reppolicy_ds_gold1`, which protects databases `orcl11` and `orcl12`, and `reppolicy_ds_gold2`, which protects only database `orcl13`.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Figure 15-5 Cascaded Replication, with Different Protection Policies on Each Recovery Appliance

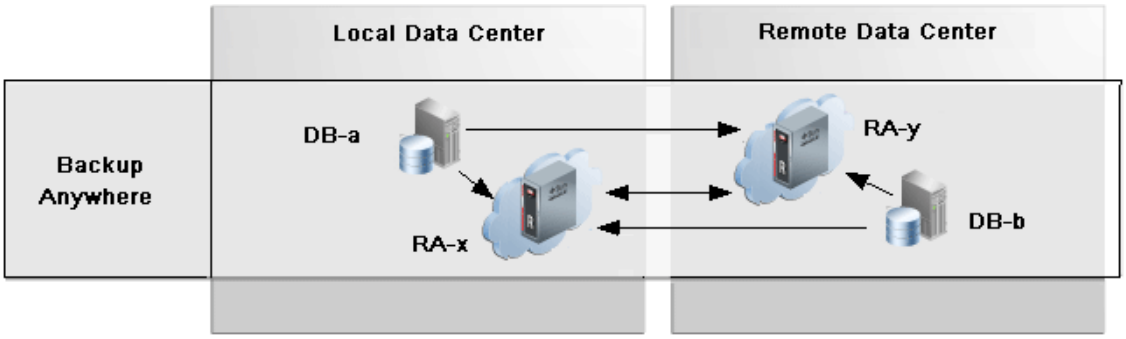


In Figure 15-5, the `reppolicy_ds_gold2` protection policy is associated with the replication server configuration `us_desm_ds_la`. `ZDLRA Des Moines` then replicates backups of `orcl13`, which is the only database protected by `reppolicy_ds_gold2`, to `ZDLRA Los Angeles`, which is the Recovery Appliance that is farthest downstream. The backups of `orcl13` that reside on `ZDLRA Los Angeles` are protected by the `reppolicy_dds_gold` policy. This configuration, in which three or more Recovery Appliances are linked in a chain, is called cascaded replication.

15.2.5 Bi-Directional Replication between Recovery Appliances

Bi-directional replication, also known as *"backup anywhere replication"*, allows two Recovery Appliances RA-x and RA-y to replicate between each other for the same (or different) protected databases. Thus, both Recovery Appliances have a full set of backups.

Copyright © 2014, 2026, Oracle and/or its affiliates.



Two Recovery Appliances RA-x and RA-y replicate between each other, and are the upstream and downstream of each other. Backups from protected databases DB-a and DB-b flow to their upstream Recovery Appliance, RA-x and RA-y respectively. New backups are replicated to the other (downstream) Recovery Appliance to synchronize the backup state across the Recovery Appliances.

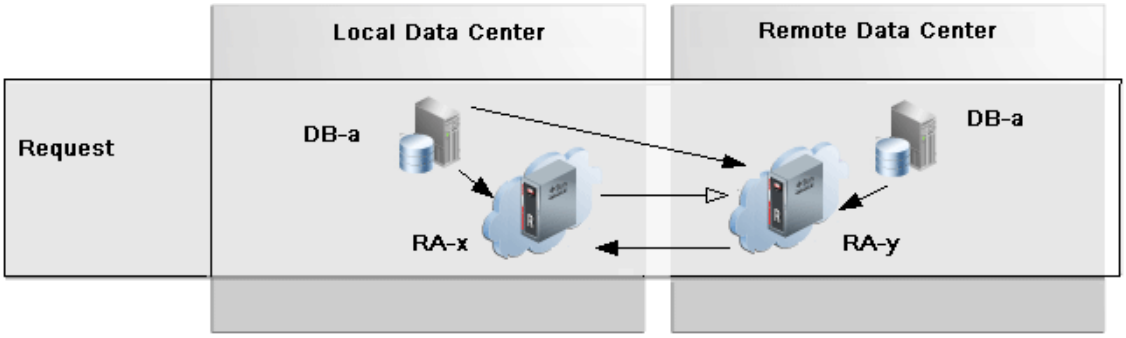
For examples of how this works in a High Availability / Disaster Recovery (HADR) configuration, refer to Replication Mode for HADR.

15.2.6 Replication Request-Only Mode

Request-only mode replication is useful for planned maintenance and reduces the amount of database backups, redo logs, and archive logs that must be transmitted between Recovery Appliances to re-establish its full recoverability of its databases at the conclusion of maintenance.

In `request_only` mode replication, the upstream (RA-x) Recovery Appliance receives the primary database backups while the downstream (RA-y) receives the standby database backups, redo logs, and archive logs. When the upstream RA-x is offline for planned maintenance as one example, the primary database redo and archive logs are redirected to the downstream RA-y, where new archived log backups are created in order to preserve database recoverability.

RA-y is not affected by RA-x outage and still receives level 1 backups from standby database as normal. Standby backups on RA-y can be used for primary database restores. When RA-x comes back online, all previously redirected backups are replicated from the downstream RA-y to the upstream RA-x to re-establish its full recoverability of its databases.



Request mode can be thought of as *"Read only replication with the ability to request backups that it missed while not running"*. Request mode is built on top of the technologies that enabled Backup Anywhere, but it is not Backup Anywhere.

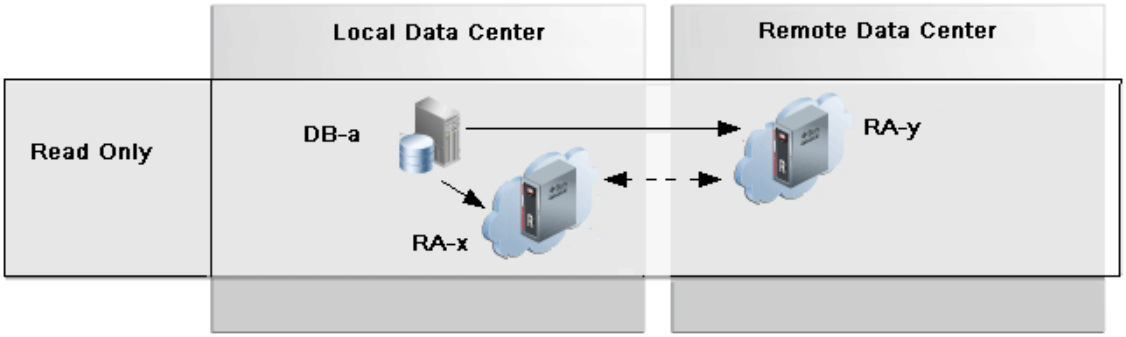
Redo and archived logs are the more critical data to retrieve from the upstream databases, because the database can hang if the local archived log directory fills up. Therefore, they are transmitted to RA-y so that they can be safely deleted by RMAN on DB-a.

Copyright © 2014, 2026, Oracle and/or its affiliates.

The command `RMAN CONFIGURE ARCHIVELOG DELETION POLICY BACKED UP 1 TIMES` establishes that when redo has been shipped and backed up on RA-y, then RA-x can safely delete the local archived logs to reclaim space. With this policy configured, the command `RMAN DELETE ARCHIVELOG` can also be used to safely reclaim local archived log space.

15.2.7 Read-Only Replication between Recovery Appliances

The `read_only` replication mode is useful when changing the destination Recovery Appliance for database backups from the original RA-x to a different RA-y while backups remain available on RA-x to RA-y for restore operations as needed. A replication server for `read_only` mode can be configured in the protection policy for RA-y and designates RA-x as the downstream. All backups for the databases in the policy that exist on the downstream are retrievable when and if needed from the upstream. After the old backups on the original RA-x have expired according to the protection policy, RA-y will have no need for the backups on RA-x allowing RA-x to be transitioned out of the backup scenario.



The important use-case for `read_only` replication is when a different Recovery Appliance is to be used as the destination for database backups, such as when commissioning new Recovery Appliances in order to balance the load or to de-commission old Recovery Appliances. This allows for the graceful transfer of backup / restore between Recovery Appliances.

15.2.8 Copy All Backups Replication

The "Copy All Backups" mode is established when a protection policy is added to the replication server with `COPYALL_BACKUPS` set to `TRUE`. It replicates all of the existing backups for the databases in that policy to a downstream Recovery Appliance. This includes backups that might be expired but not yet purged from the upstream Recovery Appliance.

On the upstream Recovery Appliance, backups are replicated in a controlled and ordered manner.

The downstream Recovery Appliance is `COPYALL_BACKUPS` aware and ingests the backups in the correct order.

Backups can be sent to either the upstream or downstream during the `COPYALL_BACKUPS` process.

On the downstream Recovery Appliance, a database can be a `COPYALL_BACKUPS` destination only once.

The `COPYALL_BACKUPS` mode is the basis for migration or moving of all database backups from one Recovery Appliance to another. The other replication modes can migrate a database to a new Recovery Appliance, but requires the full recovery window on the source upstream Recovery Appliance to be expired before the system can decommissioned.

Copyright © 2014, 2026, Oracle and/or its affiliates.

15.3 About Replication Partners and Users Accounts

As part of a disaster recovery strategy, Recovery Appliance can replicate backups to other Recovery Appliances. Also, you can offload tape archival to a replicated Recovery Appliance, thereby freeing resources on the primary Recovery Appliance. Replication is driven by protection policy, which means that all databases associated with the policy are replicated, and it is fully automatic after the initial setup.

A requirement for replication between Recovery Appliances is that all associated databases are uniquely named across all replication Recovery Appliances.

Oracle requires that you create a replication user account exclusively for use with Recovery Appliance replication, and that you create a unique replication user account for each upstream appliance within the organization.

Replication Partners

Replication is simplified through the use of *RA partners*. The replication partnership is established using RACLI that automates common administrative tasks across partners which reduces the potential for human error and the need to remotely log into each system.

RACLI establishes the connection between two Recovery Appliances.

Replication User Accounts

The replication user needs to be defined on the downstream from either `add_db/grant_db_access` or `register_tenant_user('repuser name', 1)`, which doesn't require using multi-tenancy.

Oracle recommends that the replication user account takes the form of `REPUSER_FROM_[ZDLRA_DB_NAME` or `ZDLRA_DB_LOCATION]_TO_[ZDLRA_DB_NAME or ZDLRA_DB_LOCATION]`.

For example, if two Recovery Appliances have the `DB_UNIQUE_NAME` of ZDLRA1 and ZDLRA2, then the replication user accounts could be `REPUSER_FROM_ZDLRA1_TO_ZDLRA2` and `REPUSER_FROM_ZDLRA2_TO_ZDLRA1`. Or if those same Recovery Appliances were in Florence and Vienna, then the replication user accounts could be `REPUSER_FROM_FLORENCE_TO_VIENNA` and `REPUSER_FROM_VIENNA_TO_FLORENCE`.

The replication user account is created with `racli add db_user` with `--user_type=replication`. The replication user account **should not** be used as a regular VPC user employed by protected databases to connect and send backups to the Recovery Appliance.

The following are important distinctions in the user accounts associated with replication.

`RA Partner User`
Operating System user who has a limited role enabling the creation, management, and health of a replication server used by RACLI.
One partner user is connected with one partner Recovery Appliance. It has restricted privileges based on the admin user.

`RA Replication User`
Database user that is created on downstream Recovery Appliance. Its credential is stored on upstream Recovery Appliance replication wallet.
Example: `rep_user_from_<USDB>_to_<DSDB>`

Copyright © 2014, 2026, Oracle and/or its affiliates.

Certificates
Needed for TLS secure communication between upstream and downstream RA

RA Replication Wallet
Wallet that stores all replication user credentials, downstream Recovery Appliance's certificates.

Replication Server
Replication server from upstream to downstream.
Example: `rep_server_from_<USDB>_to_<DSDB>`

15.4 RACLI Commands Relating to Replication

You can use `RACLI` commands to create and manage replication. Table 15-1 describes the main commands associated with replication and are described with more detail in the *Zero Data Loss Recovery Appliance Owner's Guide*.

Table 15-1 Principal RACLI Commands for Replication

Program Unit	Description
`racli add ra_partner`	Establishes a connection between two Recovery Appliances.
`racli add replication_server`	Adds a protection policy to a replication server.
`racli add replication_user`	Adds a database user.
`racli add replication_wallet`	Creates replication orapki wallet.
`racli alter replication_server`	Updates replication server parameters value.
`racli alter replication_user`	Changes replicatioin user password on remote cluster and updates credentials on replication wallet on local cluster.
`racli alter replication_wallet`	Updates the replication wallet password.
`racli create replication_server`	Creates replication user, replication wallet and replication server.
`racli delete replication_server`	Deletes a replication server.
`racli enable replication_management`	Allows (non-RACLI) replication configurations to be recognized by the Recovery Appliance so that RACLI can perform subsequent replication management.
`racli list ra_partner`	Shows all of the information on the partner recovery appliance including its host name.
`racli list replication_db`	List replication databases.
`racli list replication_server`	List replication server on downstream.
`racli list replication_user`	List replication user on downstream.
`racli list replication_wallet`	List replication wallet on downstream.
`racli remove replication_server`	Removes a protection policy from a replication server.
`racli remove replication_wallet`	Removes replication wallet.
`racli start replication_db`	Starts or resumes replication database.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 15-1 (Cont.) Principal RACLI Commands for Replication

Program Unit	Description
`racli start replication_server`	Starts or resumes replication server.
`racli status replication_server`	Shows status of replication server.
`racli status replication_user`	Shows status of replication users.
`racli stop replication_db`	Stops or pauses replication database.
`racli stop replication_server`	Stops or pauses replication server.
`racli update replication_server`	Update replication_server attributes in the Recovery Appliance.

See Also

RACLI Overview

15.4.1 Guidance for Replication Configuration

The letter tasks in this section are intended for guidance only. They were most likely completed already, but are provided here as a refresher and to be complete, because they are necessary for proper replication configuation.

(Guidance only) Task A: Create a Virtual Private Catalog (VPC) account

Note

This task is offered as guidance only to show how the VPC user relates to replication. VPC user creation is typically not part of the replication setup, because the VPC user creation is already completed during the initial database onboarding to the Recovery Appliance.

RMAN must connect to the Recovery Appliance catalog when backing up to a Recovery Appliance.

Copyright © 2014, 2026, Oracle and/or its affiliates.

> **Note**
>
> You can also use existing VPC users.
>
> The requirements of the VPC user password are:
>
> - At least 15 characters in total
> - At least 1 lower case letter
> - At least 1 upper case letter
> - At least 1 number
> - At least 1 special character
>
> For example: `Welcome*123456789`

In this step, you create a Virtual Private Catalog (VPC) user for a specific protected database or set of protected databases.

For one-way and two-way replication. Run from the upstream RA1:

```
racli add db_user --user_name=<REPL_VPC_USER> --user_type=vpc
```

Additionally for two-way replication. Run from the downstream RA2 with a different user name:

```
racli add db_user --user_name=<REPL_VPC_USER2> --user_type=vpc
```

```
racli list db_user
```

(Guidance only) Task B: Create protection policy.

A protection policy needs to exist on both the upstream and downstream Recovery Appliances. The upstream and downstream protection policies are independent. Four (4) default protection policies available for use.

If a suitable protection policy does not exist, create one appropriate for the replication. If this is one-way replication, create this protection policy on the upstream Recovery Appliance.

If this is two-way replication setup, run the same command on both Recovery Appliances Policy names do not need to be identical; they can be the same or different. Each Recovery Appliance operates independently.

```
racli create protection_policy --protection_policy_name=YOUR_POLICY
--storage_location_name=<YOURLOC> --recovery_window_goal=11day --
autotune_reserved_space=YES
```

Refer to `racli create protection_policy` for other appropriate arguments.

15.4.2 Configuring Replication between Recovery Appliances Using RACLI

This section describes the RACLI tasks for establishing replication between two Recovery Appliances, the upstream (RA1) and the downstream (RA2). Most of the tasks are performed on the upstream (RA1) with an `admin_user` account.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Many of the early tasks may have already been completed, if backups to the Recovery Appliance were configured and in operation. However, the tasks would be required if this were a fresh installation that configures for the first time backups and the replication.

The replication user and replication wallet are managed within `racli create replication_server` and `racli delete replication_server`. The Recovery Appliance partner user manages the replication user, replication wallet, and replication server parameters.

> **Note**
>
> If the replication was established with a version earlier than Recovery Appliance 23.1 and DBMS_RA, the recommendation is to use the RACLI commands going forward. To change replication to RACLI replication, use `racli add ra_partner` and `racli enable replication_management` as provided later in this section.

Upon completion of this, the client database sends its backups to the upstream (RA1). RA1 as part of replication sends those backups to the downstream (RA2) as well.

> **Note**
>
> The replication server in this case is on RA1. The replication user that is created is for RA2.

A partnership between Recovery Appliance allows remote management with limited privileges on RACLI commands. A partnership is mainly used for replication management, rack migration, and status checking.

Task 1: Make sure the ports are open

Make sure the ports are open. From RA1 with the path to RA2, try the SSH command:

```
ssh <RA2>
```

From RA2 with the path to RA1, try the SSH command:

```
ssh <RA1>
```

(Guidance only) Task 2: Create the `admin_user` user types needed for replication.

> **Note**
>
> This task can be skipped if the `root` user and `root` password are usable on both Recovery Appliances.

If you don't already have `admin_user` and if `root` authentication is not preferred, you need to create `admin_user` user type for replication to use.

Both Recovery Appliances require an `admin_user` (an OS user), preferably named the same.

Copyright © 2014, 2026, Oracle and/or its affiliates.

On both the upstream (RA1) and the downstream (RA2), run:

```
racli add admin_user --user_name=REPL_ADMIN_NAME
```

You are prompted for a password, which has no special requirements for the `admin_user` password.

> **Note**
>
> Remember the `admin_user` and password (`REPL_ADMIN_NAME`), because it is needed later when adding the Recovery Appliance partner.

Verify that the `admin_user` was created. On each Recovery Appliance, run::

```
racli list admin_user
```

Task 3: Create a partnership between two Recovery Appliances.

For one-way replication, create the partner connection on the upstream (RA1) to the downstrea (RA2) `--admin_user` credentials.

```
racli add ra_partner --target_host=<RA2>
[--partner_user=REPL_PARTNER_RA1 --partner_uid=UID]
--admin_user=REPL_ADMIN_NAME --admin_key=VALUE
```

- `--target_host` specifies the fully qualified domain name (FQDN) of the first compute node from the downstream Recovery Appliance. If running command from upstream RA1, the `--target_host` is RA2.
- `--partner_user` specifies the designated partner's name. This is optional and the default value is `rapartner`. The same partner user is created on both Recovery Appliances.
- `--partner_uid` specifies the numeric user id (UID) for the partner user. The UID value needs to be less than 3000 for cloud system.
- `--admin_user` specifies the administrative user, which was created in task 1. It can be either `root` or any `admin` user with access to RA2 with a password or SSH key. Required for cloud-based instances, but optional for on-premise.
- `--admin_key` specifies the `admin` user SSH key, which was obtained in task 1. Required for cloud-based instances or when an `--admin_user` is specified.

Only `--target_host` is required, which is first compute node full name of downstream Recovery Appliance (RA2). This command only needs to be run on one Recovery Appliance. It will create partner objects on both Recovery Appliances automatically

Task 4: Create a replication server.

This operation is run from the upstream Recovery Appliance (RA1). The credential information for the downstream RA2 replication user is put into the replication wallet for the upstream RA1.

Operations performed by this command:

- Add Replication user on RA2 Create Replication wallet on RA1
- TNS update

Copyright © 2014, 2026, Oracle and/or its affiliates.

- Export certificate from RA2 if RA2 is TLS enabled
- Import certificate into RA1 wallets if RA2 is TLS enabled
- Create Replication server on RA1
- If `--type=bidirectional`, all of the above operations are repeated in the opposite direction.

> **Note**
>
> This `racli create replication_server` command has a password prompt for the replication user. It creates a replication wallet if one did not exist. You do not need to create a replication user and replication wallet separately.
>
> This handles all replication objects and certificate management automatically. After run, the replication infrastructure is in place, but no data will flow yet.

```
racli create replication_server --target_host=PATH2TARGET
[--type=bidirectional]
```

- `--target_host` specifies full path to the first node of the target Recovery Appliance to be paired.
- `--type=bidirectional` specifies two-way replication. If one-way replication is needed, do not provide this argument.

The command `racli list listener` provides information on whether TCP or TCPS is used.

Example 1: Two-way replication.

```
racli create replication_server --target_host=<RA2> --type=bidirectional
```

Example 2: One-way replication with TLS enabled on only the downstream Recovery Appliance.

```
racli create replication_server --target_host=<RA2>
```

Example 2: Two-way replication with TLS enabled on only the downstream Recovery Appliance.

```
racli create replication_server --target_host=<RA2> --type=bidirectional
```

> **Note**
>
> After this task, replication is managed completely through RACLI commands include the

Copyright © 2014, 2026, Oracle and/or its affiliates.

(Guidance only) Task 5: Add protected databases.

The database needs to be added to both the upstream and downstream Recovery Appliance protection policies. The upstream should already have the database added, because it should already receive backups from the client.

At least one database has to be added to the protection policy before enabling replication.

If the database has not been assigned a protection policy by other means (DBMS_RA API or Enterprise Manager Cloud Control), then issue a command similar to.

```
racli add protected_db --db_unique_name=<yourDB>
--protection_policy_name=<YOUR_POLICY> --reserved_space=10G
```

> **Note**
>
> Each protected database can have only one (1) policy. Assigning the same database to two different protection policies is not supported. As `rasys`, you'll need to run `dbms_ra.update_db` to update the protection policies appropriately.
>
> Do **not** run `racli delete protected_db`, because this removes all of the existing backups.

(Guidance only) Task 6: Grant protected database access

> **Note**
>
> For RA 23.1 PSU prior to October 2025, the replication user must be explicitly granted access. See below.
>
> From RA23.1 PSU October 2025 onward, access is granted automatically when creating replication servers.

For one-way and two-way replication, run this on the upstream Recovery Appliance to grant access.

```
racli grant db_access --db_unique_name=test_db --
username=rep_user_from_radb01_to_radb02
```

Additionally for two-way replication, run from the downstream RA2:

```
racli grant db_access --db_unique_name=test_db --
username=rep_user_from_radb02_to_radb01
```

The command `racli list replication_server` provides the replication user names.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Task 7: Login to the client machine or Exadata Database.

The database needs to be registered and a backup has to be run. To validate whether a backup has been run or not, follow these steps.

```
racli list protected_db [ --db_unique_name=DB_NAME | --all ] [ --long ]


Created log /opt/oracle.RecoveryAppliance/log/racli_list_protected_db.log
Thu Nov  6 18:06:13 2025: Start: List Protected Database
DEMO_DB
DB UNIQUE NAME:                exadb1
DB KEY:                        5678
DBID:                          1234
TENANT NAME:                   DEFAULT_TENANT
TENANT IDENTIFIER:             0
POLICY NAME:                   REPL_PROD_RA01_to_RA02
SPACE USAGE:                   789
ACTIVE INCIDENT COUNT:         0
Thu Nov  6 18:06:13 2025: End:List Protected Database
```

Check the output:

- `SPACE USAGE` should not be 0
- `DB KEY` should not be null

Task 8: Add replication server

For one-way and two-way replication, run this on the upstream Recovery Appliance.

```
racli list replication_server
racli add replication_server
--replication_server_name=<REPL_SERVER> --protection_policy_name=<YOUR_POLICY>
```

Additionally for two-way replication, run from the downstream RA2:

```
racli list replication_server

racli add replication_server
--replication_server_name=<REPL_SERVER> --protection_policy_name=<YOUR_POLICY>
```

- `--replication_server_name` use the output from `racli list replication_server`
- `--protection_policy_name` use the protection policy specified in an earlier task.

The very first time replication is set up, issue the commands to stop and start the replication server.

```
racli stop replication_server --
replication_server_name=rep_server_from_radb02_to_radb01

racli start replication_server --replication_server_name=
rep_server_from_radb02_to_radb01
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

15.4.3 Converting Existing Replication Settings to RACLI Management

If replication was established with Recovery Appliance 21.1 or earlier with DBMS_RA commands, modifications to the configuration typically need to be performed with DBMS_RA unless you convert to RACLI (available Recovery Appliance 23.1 or later.) This requires two operations.

Add the partnership between the two Recovery Appliances. This only needs to be run on one Recovery Appliance. The `--admin_key` is not required for on-premise Recovery Appliances.

```
racli add ra_partner --target_host=<RA2> --partner_user=NAME --
partner_uid=UID
--admin_user=REPL_ADMIN_NAME [--admin_key=VALUE]
```

Enable RACLI to perform subsequent replication management.

```
racli enable replication_management --target_host=<RA2> FQDN
[--type=bidirectional] [--certificates=LIST]
```

This command migrates the original replication wallet to a new Orapki wallet and prompts the user for the new password for this wallet.

This retrieves the existing replication user, replication role, and replication server information from existing tables and updates metadata tables using the Recovery Appliance partner user. If a replication wallet doesn't exist, this creates a new replication wallet with `"orapki"` type to accept certificate in the future. Replication parameters are updated accordingly.

After this conversion, replication is managed by RACLI and the Recovery Appliance partner user.

15.4.4 Configuring Replication with a Virtual Air Gap

This section describes how to configure Recovery Appliance replication with a time-controlled limited network access, such as the Virtual Air Gap solution for a data vault scenario.

In the vault scenario, the downstream Recovery Appliance is considered *"the vault"* and has its replication network closed. The upstream Recovery Appliance has a watcher that pauses replication until it sees that downstream replication network is available. The downstream Recovery Appliance determines when its replication network is available (`open` or `limited`) and for how long (`timeout`). Once the upstream watcher sees that the downstream replication is available, it resumes replication. In the limited access mode when the timeout period expires, the downstream replication network automatically closes, causing the upstream watcher to pause replication.

Configuring the limited replication network happens after you establish a replication partnership between the upstream and downstream Recovery Appliances. You manage the replication network traffic between the two environments. The network control is implemented on the replication network only (bondeth1, replication SCAN network and VIPs).

> **Note**
>
> A pre-requisite is that the SSH is setup from the upstream to the downstream Recovery Appliance.

Copyright © 2014, 2026, Oracle and/or its affiliates.

The following tasks describe in general the sequence to create a lmited replication network (assuming the replication partnership already exists).

Task 1: Set up a watcher on the upstream Recovery Appliance

On the upstream Recovery Appliance, the watcher is turned on throught the replication server.

```
racli alter replication_server --auto_enable=true
```

Later during normal operation:

- If the upstream Recovery Appliance replication server was paused **and** if the downstream replication network became available (`open` or `limited`), then the watcher automatically resumes the replication activies to the downstream.
- If the downstream replication network became unavailable (`closed`) **and** if the upstream Recovery Appliance replication server was running, then the watcher automatically pauses the replication activies to the downstream.

Task 2: Close the downstream replication network

The replication network must initially be in closed state. From the downstream Recovery Appliance, issue the following commands.

```
racli status replication_network
racli configure replication_network --mode=closed
racli status replication_network
```

Task 3: Open the downstream replication network for a limited time

Issue the following command with a `<TIMEOUT>` in minutes of your chosing. This opens the downstream replication network for the specified timeout period. After this period expires, the replication network is automatically closed.

```
racli configure replication_network --mode=limited [--timeout=<TIMEOUT>]
```

The `timeout` is in minutes and defaults to 60 minutes.

Normal Limited Access or Vault Operations

Once task 1 and 2 are completed, the vault configuration is complete.

Thereafter, task 3 is repeated on the downstream Recovery Appliance (the vault) on whatever manual or automated schedule is appropriate for the your application.

- Whenever the downstream replication network became available (`open` or `limited`), the paused upstream Recovery Appliance replication server automatically resumes the replication activies to the downstream.
- When the downstream replication network became unavailable (`closed`), the upstream Recovery Appliance replication server pauses the replication activies to the downstream.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Turn off limited access and turn on normal replication

To turn on normal replication between an upstream Recovery Appliance and the downstream, you open the downstream replication network.

```
racli configure replication_network --mode=open
```

If you no longer need limited access capabilities, you can stop the watch for the replication server on the upstream Recovery Appliance.

```
racli alter replication_server --auto_enable=false
```

If you turn off the upstream watcher (without opening the downstream replication network), the result may be replication log errors for the periods when the downstream replication network isn't available.

15.5 Accessing the Replication Page in Cloud Control

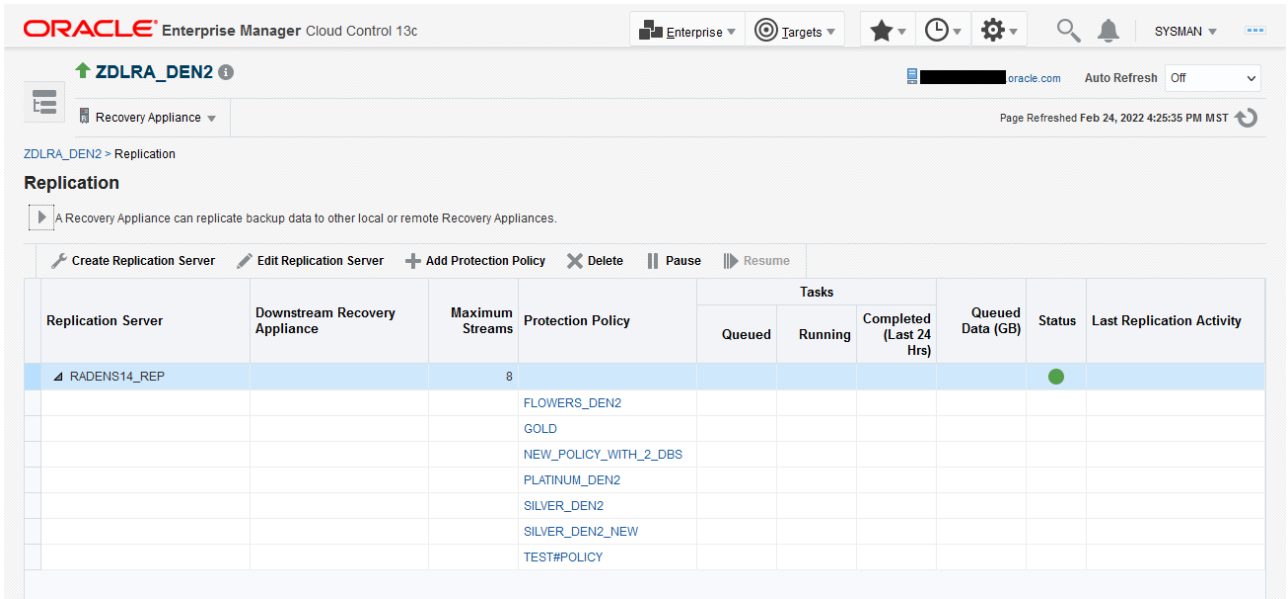
The Replication page in Cloud Control is the recommended interface for configuring Recovery Appliance replication.

To access the Replication page:

1. Access the Recovery Appliance Home page, as described in "Accessing the Recovery Appliance Home Page".
2. From the **Recovery Appliance** menu, select **Replication**.

   The Replication page appears, as shown in Figure 15-6.

Figure 15-6 Replication Page



In the figure above, the replication server named `RADENS14_REP` is already configured. The Status column shows that it is available.

Copyright © 2014, 2026, Oracle and/or its affiliates.

15.5.1 Configuring Recovery Appliance Replication Using Cloud Control

This section describes how to configure Recovery Appliance replication using the Replication page in Cloud Control.

Prerequisites

Your environment must meet the following prerequisites:

- The upstream and downstream Recovery Appliance can communicate with each other over the network.
- Every protected database whose backup data will be replicated must be enrolled with the upstream Recovery Appliance.
- The downstream Recovery Appliance must be started and configured to receive backups.

Assumptions

Assume that the following statements are true of your Recovery Appliance environment:

- The protected databases `orcl11` and `orcl12` back up to upstream Recovery Appliance `ZDLRA Boston`.
- You want `ZDLRA Boston` to replicate to downstream Recovery Appliance `ZDLRA_DEN2`.
- A replication user account named `repuser_from_boston` exists on the *downstream* Recovery Appliance (`ZDLRA_DEN2`).
- A virtual private catalog account named `vpc_boston1` exists on the *upstream* Recovery Appliance (`ZDLRA Boston`).
- You know the credentials for the operating system user who owns the upstream Recovery Appliance (`ZDLRA Boston`) database installation.

To configure Recovery Appliance replication:

1. Access the Recovery Appliance Home page on the *upstream* Recovery Appliance. Refer to "Accessing the Recovery Appliance Home Page".
2. From the **Recovery Appliance** menu on the *upstream* Recovery Appliance, select **Protection Policies** to create one for replication.

   In this example, you access the **Create Protection Policy** page for `ZDLRA Boston`, which is the upstream Recovery Appliance.

   If required, enter your login credentials, and then click **Login**.

   The Protection Policies page appears.
3. Create a **replication** protection policy. This is a normal protection policy that is later tied to replication, and is presented in Creating a Protection Policy.

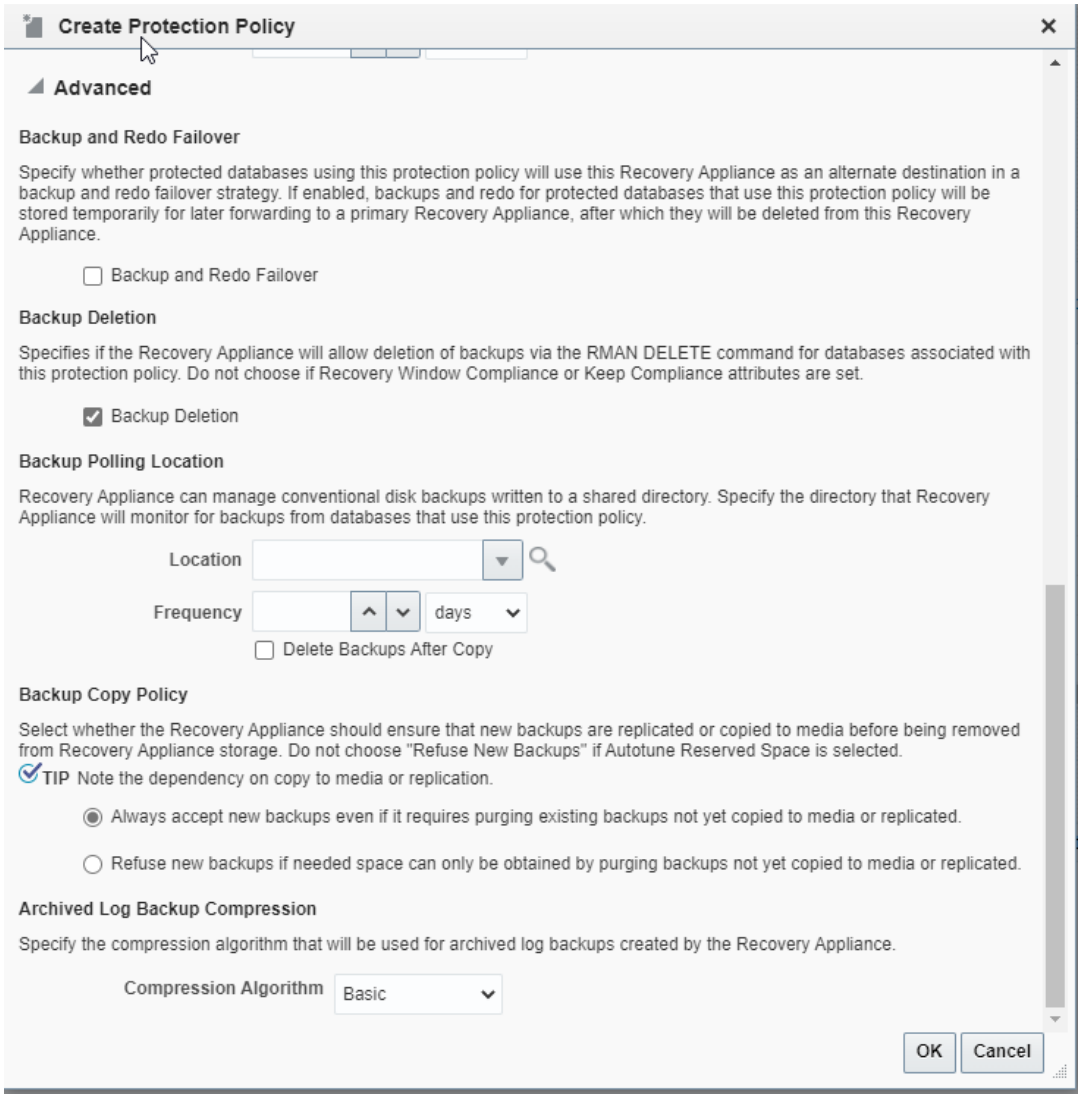
Copyright © 2014, 2026, Oracle and/or its affiliates.

Figure 15-7 Create Protection Policy Page



Copyright © 2014, 2026, Oracle and/or its affiliates.

Figure 15-8 Protection Policy Advanced Parameters



In this example, you create a policy named `reppolicy_ds_gold`.

4. From the **Recovery Appliance** drop-down menu on the *upstream* Recovery Appliance, select **Protected Database**.

   Select **Add** from the **Protected Database** tool bar. It shows the **Add Protected Databases** dialog box.

5. Select **Add** from this dialog box, which provides a **Select Targets** tool to **Search** for databases.

   After a database is selected, it appears in the **Database** table. You can repeat this for other databases.

6. After the databases have been selected in **Add Protected Databases** dialog box, select the appropriate replication policy, such as `reppolicy_ds_gold` in this example.

   When finished, select **Next** creates a job for the databases in the replication protection policy.

Copyright © 2014, 2026, Oracle and/or its affiliates.

7. From the **Recovery Appliance** drop-down menu on the *upstream* Recovery Appliance, select **Replication**.

   If the Recovery Appliance Login page appears, enter your login credentials and then click **Login**.

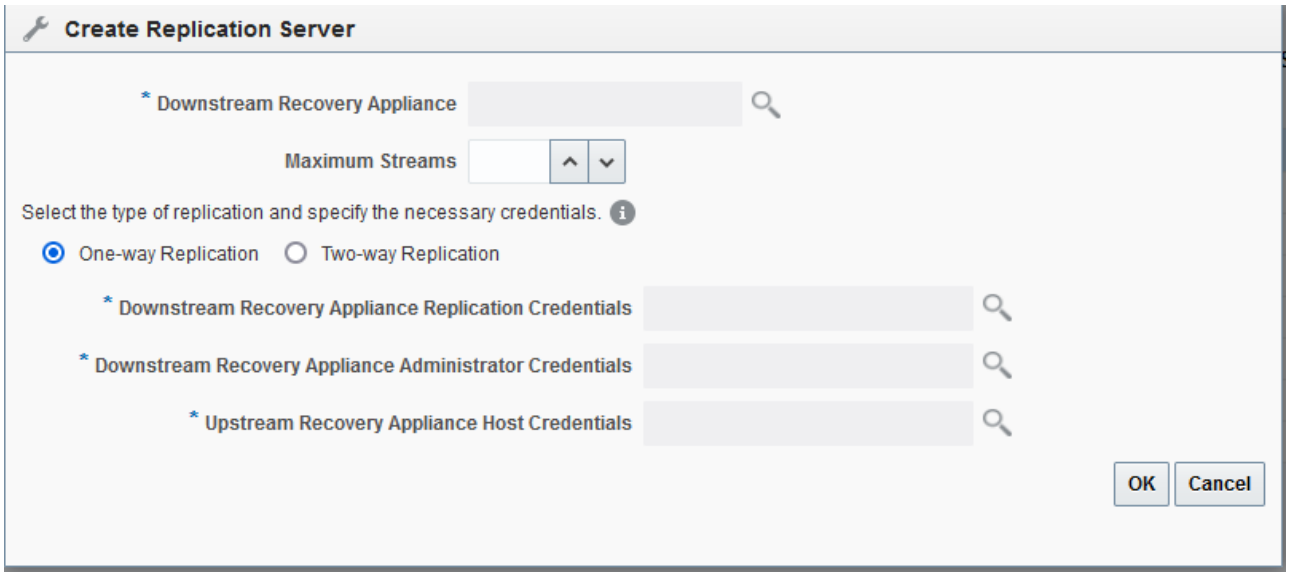
   The Replication page appears, as shown in Figure 15-6.

   In the preceding example, the replication server was named `ZDLRA9_REP`.

8. Select **Create Replication Server** on the **Replication** page.

   The Create Replication Server page appears.

Figure 15-9 Create Replication Server Page



9. Enter values as follows, and then click **OK**:

   - In the **Downstream Recovery Appliance** field, click the magnifying glass, and then from the list of discovered targets, select the Recovery Appliance that you want to configure in the downstream role.

     For example, select **ZDLRA_DEN2**.

   - In the **Downstream Recovery Appliance Database Credentials** section, specify credentials for a virtual private catalog account on the downstream Recovery Appliance.

     > **Note**
     >
     > This catalog account must have been granted permission to manage the replicated backups on the downstream Recovery Appliance. See racli add db_user.

     For example, enter `vpc_den2`.

   - In the **Upstream Recovery Appliance Database Credentials** section, specify credentials for the operating system user who owns the upstream Recovery Appliance database installation.

Copyright © 2014, 2026, Oracle and/or its affiliates.

An Information message appears showing the job submission ID.

10. After the new Replication Server appears in the list on the **Replication** page, select **Add Protection Policy**.

    In this example, you created a policy named `reppolicy_ds_gold` to be added to the replication server.

Once the replication server is configured on the upstream Recovery Appliance, the policies and configuration are handled for the downstream Recovery Appliance.

15.6 DBMS_RA Procedures Relating to Replication

You can use the `DBMS_RA` package to create and manage replication. Table 15-2 describes the principal program units relating to replication.

Table 15-2 Principal Procedures Relevant for Replication

Program Unit	Description
CREATE_REPLICATION_SERVER	Creates a replication server configuration that specifies a downstream Recovery Appliance to which this Recovery Appliance replicates backups.
DELETE_REPLICATION_SERVER	Deletes a replication server configuration.
ADD_REPLICATION_SERVER	Adds a replication server configuration to the protection policy that was created by the `CREATE_REPLICATION_SERVER` procedure.
REMOVE_REPLICATION_SERVER	Removes a replication server configuration from the protection policy that was created by the `CREATE_REPLICATION_SERVER` procedure.
ADD_DB	Adds a database to the protection policy.
CREATE_PROTECTION_POLICY	Creates a protection policy. To enable replication for databases assigned to this policy, you must associate a replication server configuration with this policy by running `ADD_REPLICATION_SERVER`.
UPDATE_DB	Updates the properties of a protected database.

See Also

DBMS_RA Package Reference

15.6.1 Configuring Recovery Appliance for Replication Using DBMS_RA

This section explains how to configure replication using command-line tools. The basic work flow is as follows:

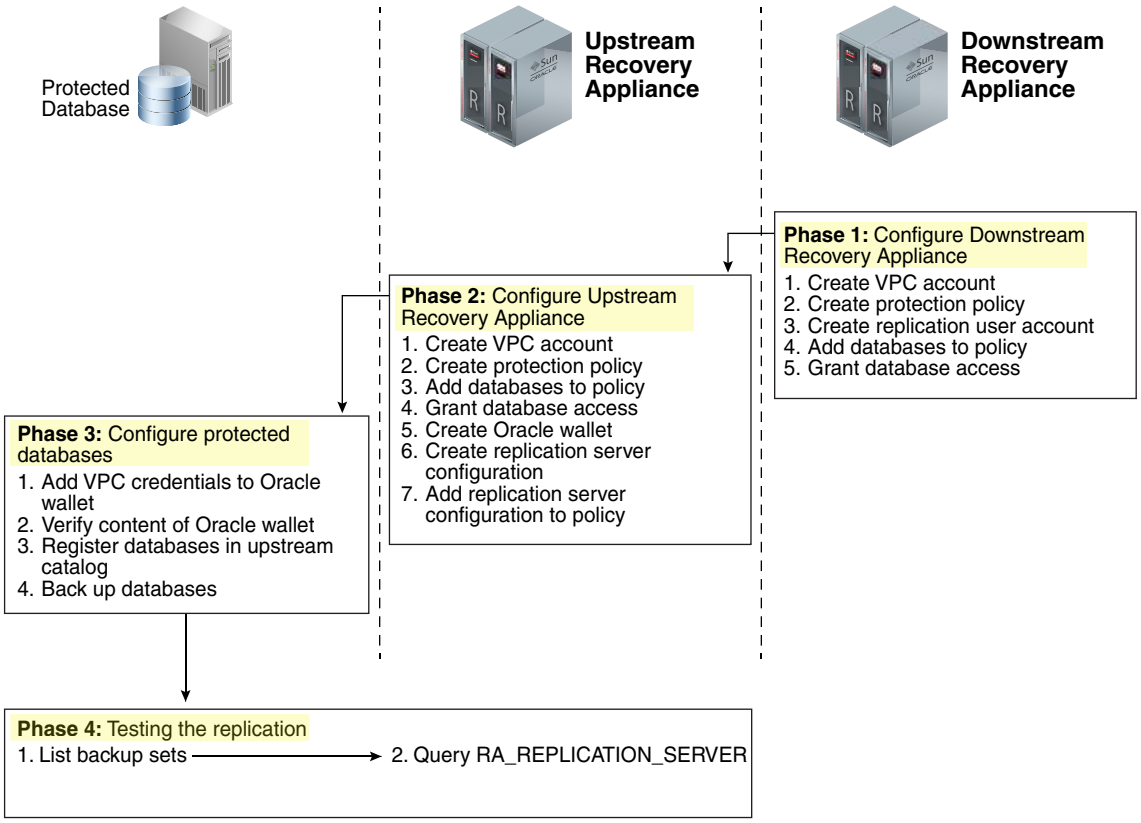
1. Configure the downstream Recovery Appliance, as described in "Configuring a Downstream Recovery Appliance for Replication Using DBMS_RA".
2. Configure the upstream Recovery Appliance, as described in "Configuring an Upstream Recovery Appliance for Replication Using DBMS_RA".
3. Configure the protected databases involved in the replication, as described in "Configuring a Protected Database for Recovery Appliance Replication".

Copyright © 2014, 2026, Oracle and/or its affiliates.

4. Test the replication, as described in "Testing a Recovery Appliance Replication Server Configuration".

Figure 15-10 is a graphic illustration of the configuration phases.

Figure 15-10 Overview of Manual Configuration for Replication



15.6.1.1 Assumptions for the Replication Examples

In the replication tasks that follow, assume that the following conditions are true:

- You back up databases `orcl11` and `orcl12` to a Recovery Appliance named `ZDLRA Boston` that you want to configure in the *upstream* replication role.
- You intend to use `ZDLRA Des Moines` as the *downstream* Recovery Appliance.
- On the downstream Recovery Appliance, you intend to create a Recovery Appliance user account named `repuser_from_boston`. This account is the replication user account.

> **Note**
>
> The naming convention for this account uses the Recovery Appliance from where the backups will be replicated—in this case, `ZDLRA Boston`. In names of the protection policies in our examples, we use `us` for upstream and `ds` for downstream.

Copyright © 2014, 2026, Oracle and/or its affiliates.

- On the downstream Recovery Appliance, you intend to create a protection policy named `reppolicy_ds_gold`. This policy is exclusively for use by replication.
- On the downstream Recovery Appliance, you intend to create a virtual private catalog account named `vpc_des_moines1`. RMAN uses this account to back up and restore databases `orcl11` and `orcl12`.
- On the upstream Recovery Appliance, you intend to create a protection policy named `reppolicy_us_gold`. This policy is exclusively for use by replication.
- On the upstream Recovery Appliance, you intend to create a virtual private catalog account named `vpc_boston1`. RMAN uses this account to back up and restore databases `orcl11` and `orcl12`.

15.6.1.2 Configuring a Downstream Recovery Appliance for Replication Using DBMS_RA

This section explains how to configure a downstream Recovery Appliance.

> **Note**
>
> When a Recovery Appliance has both the upstream and downstream roles, these instructions pertain to the role of a downstream Recovery Appliance only.

Task 1: Create a virtual private catalog account on the downstream Recovery Appliance

When backing up or restoring protected databases, RMAN uses this account to connect to the recovery catalog on the downstream Recovery Appliance.

This task assumes that you want to create a virtual private catalog account named `vpc_des_moines1` on the downstream Recovery Appliance.

To create virtual private catalog account:

- Follow the instructions in racli add db_user.

  For example, execute the following statement to create user account `vpc_des_moines1`:

  ```
  # ./racli add db_user --user_name=vpc_des_moines1 --user_type=vpc
  ```

  Enter the password for `vpc_des_moines1` user when prompted.

> **See Also**
>
> *Oracle Database Backup and Recovery User's Guide* to learn more about virtual private catalogs

Task 2: Create a replication protection policy on the downstream Recovery Appliance

To create a protection policy specifying recovery windows and other properties of backups replicated to this downstream Recovery Appliance, execute `DBMS_RA.CREATE_PROTECTION_POLICY`.

Copyright © 2014, 2026, Oracle and/or its affiliates.

This task assumes that you create a `reppolicy_ds_gold` policy to protect the `orcl11` and `orcl12` databases. You will later associate this policy with a Recovery Appliance.

To create a replication protection policy:

1. With SQL*Plus or SQL Developer, connect to the downstream Recovery Appliance database as `RASYS`.
2. Create a protection policy with the `DBMS_RA.CREATE_PROTECTION_POLICY` procedure.

   For example, execute the following PL/SQL program:

```
BEGIN
  DBMS_RA.CREATE_PROTECTION_POLICY (
    protection_policy_name => 'reppolicy_ds_gold',
    description            => 'For protected dbs in gold tier',
    storage_location_name  => 'delta',
    recovery_window_goal   => INTERVAL '28' DAY,
    guaranteed_copy        => 'NO');
END;
```

See Also

- racli add db_user
- "CREATE_PROTECTION_POLICY" for definitions of procedure arguments

Task 3: Create a replication user account on the downstream Recovery Appliance

When you configure a *downstream* Recovery Appliance to replicate backups for a protected database, you must create a replication user account that the *upstream* Recovery Appliance uses to log in to this downstream Recovery Appliance. The credentials for the user on the *downstream* Recovery Appliance are stored in the Oracle wallet of the *upstream* Recovery Appliance (see "Task 5: Create an Oracle wallet on the upstream Recovery Appliance").

Note

For ease of administration, Oracle recommends that you create a replication user account exclusively for use with Recovery Appliance replication, and that you create a separate replication user account for each upstream appliance.

This task assumes that you want to create an account named `repuser_from_boston` that the upstream Recovery Appliance uses to authenticate on this Recovery Appliance.

To create a replication user account:

1. With SQL*Plus or SQL Developer, connect to the downstream Recovery Appliance database as `SYSTEM` or any user with the `DBA` role.
2. Create the replication user account.

   For example, execute the following SQL statements to create the `repuser_from_boston` database user account and grant it `CREATE SESSION` privileges:

```
# ./racli add db_user --user_name=repuser_from_boston --user_type=vpc
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

> **Note**
>
> Oracle recommends that you use a highly complex password to enhance security. You add this password and user name to the Oracle wallet in a subsequent step. After you save these credentials in the wallet, you will not need to enter the password manually again.

> **See Also**
>
> *Oracle Database Security Guide* to learn how to create database user accounts

Task 4: Add databases to the protection policy on the downstream Recovery Appliance

To add the protected databases to the replication protection policy, execute `DBMS_RA.ADD_DB`. You must also specify the amount of disk space reserved for each protected database.

This task assumes that you want to add databases `orcl11` and `orcl12` to the `reppolicy_ds_gold` protection policy that you created in Task 2: Create a replication protection policy on the downstream Recovery Appliance and allocate 128 GB of reserved space for each protected database.

To add databases to a protection policy:

1. With SQL*Plus or SQL Developer, connect to the downstream Recovery Appliance database as `RASYS`.
2. Add metadata for each protected database using the `DBMS_RA.ADD_DB` procedure.

   For example, execute the following PL/SQL programs:

```
BEGIN
  DBMS_RA.ADD_DB (
    db_unique_name         => 'orcl11',
    protection_policy_name => 'reppolicy_ds_gold',
    reserved_space         => '128G');
END;
BEGIN
  DBMS_RA.ADD_DB (
    db_unique_name         => 'orcl12',
    protection_policy_name => 'reppolicy_ds_gold',
    reserved_space         => '128G');
END;
```

> **See Also**
>
> "ADD_DB"

Task 5: Grant database access on the downstream Recovery Appliance

Execute `DBMS_RA.GRANT_DB_ACCESS` to grant protected database access to the following database accounts:

- The virtual private catalog account created in "Task 1: Create a virtual private catalog account on the downstream Recovery Appliance"

Copyright © 2014, 2026, Oracle and/or its affiliates.

- The replication user account created in "Task 3: Create a replication user account on the downstream Recovery Appliance"

To grant protected database access to the replication and catalog accounts:

1. With SQL*Plus or SQL Developer, connect to the downstream Recovery Appliance database as `RASYS`.
2. For each protected database that will send backups to the upstream Recovery Appliance that must authenticate with this account, grant privileges to the replication user.

   The following example grants the replication user `repuser_from_boston` the required privileges on protected databases `orcl11` and `orcl12`:

```
BEGIN
  DBMS_RA.GRANT_DB_ACCESS (
    username       => 'repuser_from_boston',
    db_unique_name => 'orcl11');
END;
BEGIN
  DBMS_RA.GRANT_DB_ACCESS (
    username       => 'repuser_from_boston',
    db_unique_name => 'orcl12');
END;
```

3. For each protected database on each upstream Recovery Appliance that will authenticate with this account, grant privileges to the virtual private catalog account.

   The following example grants the recovery catalog account `vpc_des_moines1` the required privileges on protected databases `orcl11` and `orcl12`:

```
BEGIN
  DBMS_RA.GRANT_DB_ACCESS (
    username       => 'vpc_des_moines1',
    db_unique_name => 'orcl11');
END;
BEGIN
  DBMS_RA.GRANT_DB_ACCESS (
    username       => 'vpc_des_moines1',
    db_unique_name => 'orcl12');
END;
```

See Also

"GRANT_DB_ACCESS"

15.6.1.3 Configuring an Upstream Recovery Appliance for Replication Using DBMS_RA

This section explains how to configure an upstream Recovery Appliance. This section assumes that you have completed the steps in "Configuring a Downstream Recovery Appliance for Replication Using DBMS_RA".

Copyright © 2014, 2026, Oracle and/or its affiliates.

> **Note**
>
> When a Recovery Appliance has both the upstream and downstream roles, these instructions pertain to the upstream role only.

Task 1: Create a virtual private catalog account on the upstream Recovery Appliance

When backing up protected databases, RMAN uses this account to connect to the recovery catalog on the upstream Recovery Appliance.

This section assumes that you want to create a virtual private catalog account named `vpc_boston1` on the upstream Recovery Appliance.

To create virtual private catalog account:

- Follow the instructions in "racli add db_user".

  For example, execute the following statement to create user account `vpc_boston1`:

  ```
  # ./racli add db_user --user_name=vpc_boston1 --user_type=vpc
  ```

  Enter the password for `vpc_boston1` user when prompted.

> **See Also**
>
> *Oracle Database Backup and Recovery User's Guide* to learn more about virtual private catalogs

Task 2: Create a protection policy on the upstream Recovery Appliance

Execute `DBMS_RA.CREATE_PROTECTION_POLICY` to create a protection policy to specify the disk recovery windows and other properties of backups to this upstream Recovery Appliance. The upstream Recovery Appliance replicates these backups to its downstream Recovery Appliance.

This task assumes that you create a `reppolicy_us_gold` policy to protect the `orcl11` and `orcl12` databases. In the next task, you associate this protection policy with the protected databases.

To create a protection policy for Recovery Appliance replication:

1. With SQL*Plus or SQL Developer, connect to the upstream Recovery Appliance metadata database as `RASYS`.
2. Create each protection policy with the `DBMS_RA.CREATE_PROTECTION_POLICY` procedure.

   For example, execute the following PL/SQL program:

   ```
   BEGIN
     DBMS_RA.CREATE_PROTECTION_POLICY (
       protection_policy_name => 'reppolicy_us_gold',
       description            => 'For protected dbs in gold tier',
       storage_location_name  => 'delta',
       recovery_window_goal   => INTERVAL '28' DAY,
       guaranteed_copy        => 'NO');
   END;
   ```


Copyright © 2014, 2026, Oracle and/or its affiliates.

> **See Also**
>
> - "Creating a Protection Policy"
> - "CREATE_PROTECTION_POLICY" for definitions of procedure arguments

Task 3: Add databases to the protection policy on the upstream Recovery Appliance

To add the protected databases to the replication protection policy, execute the `DBMS_RA.ADD_DB` procedure. You must also specify the amount of disk space reserved for each protected database.

This task assumes that you want to add databases `orcl11` and `orcl12` to the `reppolicy_us_gold` protection policy that you created in "Task 2: Create a protection policy on the upstream Recovery Appliance", and allocate 128 GB of reserved space for each protected database.

To add databases to a protection policy:

1. With SQL*Plus or SQL Developer, connect to the upstream Recovery Appliance metadata database as `RASYS`.
2. Add metadata for each protected database using the `DBMS_RA.ADD_DB` procedure.

   For example, execute the following PL/SQL programs:

```
BEGIN
  DBMS_RA.ADD_DB (
    db_unique_name         => 'orcl11',
    protection_policy_name => 'reppolicy_us_gold',
    reserved_space         => '128G');
END;
BEGIN
  DBMS_RA.ADD_DB (
    db_unique_name         => 'orcl12',
    protection_policy_name => 'reppolicy_us_gold',
    reserved_space         => '128G');
END;
```

> **See Also**
>
> - "Enrolling Protected Databases"
> - "ADD_DB"

Task 4: Grant database access to the virtual private catalog account on the upstream Recovery Appliance

To grant protected database access to the *upstream* catalog account created in "Task 1: Create a virtual private catalog account on the upstream Recovery Appliance", execute `DBMS_RA.GRANT_DB_ACCESS`. This step makes it possible for RMAN to connect to the recovery catalog when it backs up or restores the protected databases.

To grant protected database access to the virtual private catalog:

Copyright © 2014, 2026, Oracle and/or its affiliates.

1. With SQL*Plus or SQL Developer, connect to the upstream Recovery Appliance metadata database as `RASYS`.
2. For each protected database whose backups will be replicated, grant privileges to the virtual private catalog account.

   The following example grants the catalog account `vpc_boston1` the required privileges on protected databases `orcl11` and `orcl12`:

```
BEGIN
  DBMS_RA.GRANT_DB_ACCESS (
    username       => 'vpc_boston1',
    db_unique_name => 'orcl11');
END;
BEGIN
  DBMS_RA.GRANT_DB_ACCESS (
    username       => 'vpc_boston1',
    db_unique_name => 'orcl12');
END;
```

See Also

- "Enrolling Protected Databases"
- "GRANT_DB_ACCESS"

Task 5: Create an Oracle wallet on the upstream Recovery Appliance

On the upstream Recovery Appliance, use the `mkstore` utility to create an Oracle auto-login wallet and add the replication user credentials created in "Task 3: Create a replication user account on the downstream Recovery Appliance". The upstream Recovery Appliance requires these credentials when it logs in to a downstream appliance. Each stored credential contains the name and verifier of a Recovery Appliance user account.

Note

If an existing wallet is an auto-login wallet (one that does not require you to enter a password each time the wallet is accessed), then you may use it. An Oracle wallet has a file extension of `*.sso`. To use an existing Oracle wallet, skip Step 2 below.

This task assumes the following:

- You want to create the Oracle wallet used for replication in the `/dbfs_repdbfs/REPLICATION` directory on the upstream Recovery Appliance host.
- You want to add credentials for replication user `repuser_from_boston`.

To create an Oracle Wallet on the upstream Recovery Appliance:

1. Log in to the upstream Recovery Appliance host as the operating system user who installed Recovery Appliance or as a member of that user's operating system group.
2. To create the Oracle wallet, run the following command, where *`wallet_location`* is an existing directory on the upstream Recovery Appliance in which to store the wallet:

```
orapki -wrl wallet_location -createALO
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

For example, the following command creates an auto-login wallet in the `/dbfs_repdbfs/REPLICATION` directory:

```
orapki -wrl file:/dbfs_repdbfs/REPLICATION -createALO
```

The `mkstore` utility creates a file named `cwallet.sso` in the designated location.

3. To add the credentials, run the following command:

```
mkstore -wrl wallet_location -createCredential serv_name ds_rep_user pwd
```

The placeholders are defined as follows:

- *wallet_location* is the directory in which to create the wallet. The directory must exist.
- *serv_name* is an Oracle network service name that you use in an EZ Connect descriptor to identify the downstream Recovery Appliance on an Oracle network.
- *ds_rep_user* is the user name of the replication user account on the downstream Recovery Appliance.
- *pwd* is the secure password of the replication user on the downstream Recovery Appliance.

For example, the following command adds credentials for the net service name `radsm01repl-scan.acme.com` using port `1522` and a database name of `zdlradsm`, and the replication user name `repuser_from_boston`:

```
mkstore -wrl file:/dbfs_repdbfs/REPLICATION -createCredential \
"radsm01repl-scan.acme.com:1522/zdlradsm" "repuser_from_boston" "pwd"
```

4. Verify that credentials were properly added for all users by running the following command, which lists the credentials in the Oracle wallet (no passwords or verifiers are displayed):

```
mkstore -wrl wallet_location -listCredential
```

For example, the following command lists the credentials in the Oracle wallet stored in `/dbfs_repdbfs/REPLICATION`:

```
mkstore -wrl file:/dbfs_repdbfs/REPLICATION -listCredential

Oracle Secret Store Tool : Version 12.1.0.1 Copyright (c) 2004, 2012, Oracle and/or
its affiliates. All rights reserved.
List credential (index: connect_string username)
1: radsm01repl-scan1.acme.com:1522/zdlradsm repuser_from_boston
```

See Also

- *Oracle Database Net Services Administrator's Guide* for the location of `tnsnames.ora`
- *Oracle Database Net Services Administrator's Guide* to learn more about net service names

Task 6: Create the replication server configuration on the upstream Recovery Appliance

For each downstream Recovery Appliance to which this upstream Recovery Appliance will replicate, create a replication server configuration by executing `DBMS_RA.CREATE_REPLICATION_SERVER`.

Copyright © 2014, 2026, Oracle and/or its affiliates.

> **⚠ Caution**
>
> If you run `CREATE_REPLICATION_SERVER` on the upstream Recovery Appliance *before* the downstream Recovery Appliance has added the databases to a protection policy (`ADD_DB`) and granted database access (`GRANT_DB_ACCESS`), then an `ORA-*` error can result.

This task assumes the following:

- You want to create a replication server configuration named `zdlradsm_rep`.

  > **ⓘ Note**
  >
  > The replication server configuration name is arbitrary. However, Oracle recommends that you use the service name of the downstream Recovery Appliance, which is also the database name (`zdlradsm` in this example) followed by `_rep`.

- You want the upstream Recovery Appliance to log in to its downstream Recovery Appliance using the replication account `repuser_from_boston`. You created this account in "Task 3: Create a replication user account on the downstream Recovery Appliance".
- The configuration uses the net service name `radsm01repl-scan.acme.com:1522/zdlradsm` that you stored in the Oracle wallet created in "Task 5: Create an Oracle wallet on the upstream Recovery Appliance".
- The Oracle wallet is stored in `/dbfs_repdbfs/REPLICATION`.
- The file name of the Recovery Appliance Backup Module, which is preinstalled on every Recovery Appliance, is `libra.so`. The module functions as an SBT media management library. RMAN references this module when allocating or configuring a channel for backup to the Recovery Appliance (see "Configuring a Protected Database for Recovery Appliance Replication").

To create a replication server configuration:

1. With SQL*Plus or SQL Developer, connect to the upstream Recovery Appliance metadata database as `RASYS`.
2. Run the `DBMS_RA.CREATE_REPLICATION_SERVER` procedure for each downstream Recovery Appliance.

   The following example creates the replication server configuration named `zdlradsm_rep` for the downstream Recovery Appliance named `ZDLRA Des Moines`:

   ```
   BEGIN
     DBMS_RA.CREATE_REPLICATION_SERVER (
       replication_server_name => 'zdlradsm_rep',
       sbt_so_name      => 'libra.so',
       catalog_user_name       => 'RASYS',
       wallet_alias            => 'radsm01repl-scan.acme.com:1522/zdlradsm',
       wallet_path             => 'file:/dbfs_repdbfs/REPLICATION');
   END;
   ```

3. Confirm the creation of the replication server configuration.

   For example, run the following query:

Copyright © 2014, 2026, Oracle and/or its affiliates.

```
SELECT COUNT(*) should_be_one
FROM   RA_REPLICATION_CONFIG
WHERE  REPLICATION_SERVER_NAME = 'ZDLRADSM_REP';

SHOULD_BE_ONE
-------------
1
```

If the configuration was created correctly, then the return value is `1`.

> **See Also**
>
> - "CREATE_REPLICATION_SERVER" for procedure argument descriptions
> - *Zero Data Loss Recovery Appliance Protected Database Configuration Guide* to learn more about the Recovery Appliance Backup Module
> - *Oracle Database Backup and Recovery User's Guide* for a list of valid client configuration file parameters and their definitions

Task 7: Associate the upstream Recovery Appliance with a protection policy

Specify the downstream Recovery Appliances to which each protected database replicates by assigning the replication server configuration to a protection policy. When this task is completed, Recovery Appliance replication is enabled.

> **Note**
>
> You can assign multiple replication server configurations to a protection policy.

This task assumes the following:

- You want to use the replication server configuration named `zdlradsm_rep`, which you created in "Task 6: Create the replication server configuration on the upstream Recovery Appliance".
- You want to add the replication server configuration to protection policy `reppolicy_us_gold`, which you created in "Task 2: Create a protection policy on the upstream Recovery Appliance".

To associate a replication server configuration with a protection policy:

1. Ensure you are connected to the Recovery Appliance metadata database as the Recovery Appliance administrator.
2. Run the `DBMS_RA.ADD_REPLICATION_SERVER` procedure for each combination of protection policy and replication server configuration.

   For example, execute the following PL/SQL program:

```
BEGIN
  DBMS_RA.ADD_REPLICATION_SERVER (
   replication_server_name => 'zdlradsm_rep',
   protection_policy_name  => 'reppolicy_us_gold');
END;
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

> **See Also**
>
> "ADD_REPLICATION_SERVER"

15.6.1.4 Configuring a Protected Database for Recovery Appliance Replication

Each protected database that participates in a Recovery Appliance replication environment must be correctly configured. For example, for each protected database, you must:

- Add the Oracle wallet credentials for the virtual private catalog owner on the upstream *and* downstream Recovery Appliances to the Oracle wallet.

  > **Note**
  >
  > The replication configuration does not require you to add the downstream credentials. However, if the upstream Recovery Appliance were inaccessible, and if RMAN tried to restore backups from the downstream Recovery Appliance, then RMAN would need to connect directly to the virtual private catalog in the downstream Recovery Appliance. In this case, the Oracle wallet would require the downstream credentials.

- Verify the content of the Oracle wallet.
- Register the database in the virtual private catalog of the *upstream* Recovery Appliance.
- Back up the protected database, making sure to specify the correct Oracle wallet location when allocating the RMAN channel.

To learn how to configure protected databases, see *Zero Data Loss Recovery Appliance Protected Database Configuration Guide*.

15.6.1.5 Testing a Recovery Appliance Replication Server Configuration

For every protected database involved in a replication scheme, use the following procedure to test replication from an upstream Recovery Appliance to all downstream Recovery Appliances. You can repeat this procedure to test each replication path of a complex replication topology.

This section assumes the following:

- You want to test the replication of backups of `orcl11` from `ZDLRA Boston`, which is the upstream Recovery Appliance, to `ZDLRA Des Moines`, which is the downstream Recovery Appliance.
- `ZDLRA Boston` also backs up to tape.

To test the replication of a protected database:

1. Start RMAN, and connect to a protected database as `TARGET`, and the virtual private catalog on the upstream Recovery Appliance as `CATALOG`.

   For example, enter the following command at the system prompt to connect to `orcl11` as `TARGET` and `zdlra_boston` as `CATALOG`:

   ```
   rman TARGET ra_admin@orcl11 CATALOG /@zdlra01bosingest-scan1.acme.com:1521/
   zdlrabos:dedicated
   ```


Copyright © 2014, 2026, Oracle and/or its affiliates.

2. List the backup sets, and confirm that the backups exist on the upstream and downstream Recovery Appliances.

   For example, run the following command (sample output included):

```
RMAN> LIST BACKUPSET;
.
.
.

BS Key  Size
------- ----------
54746   224.25M

  List of Archived Logs in backup set 54746
  Thrd Seq     Low SCN    Low Time            Next SCN   Next Time
  ---- ------- ---------- ------------------- ---------- ---------
  1    17854   153525644  2014/07/01 12:59:40 153545145  2014/07/01 13:00:34
  1    17855   153545145  2014/07/01 13:00:34 153564529  2014/07/01 13:01:36
  1    17856   153564529  2014/07/01 13:01:36 153585644  2014/07/01 13:02:26
  1    17857   153585644  2014/07/01 13:02:26 153606722  2014/07/01 13:03:18
  1    17858   153606722  2014/07/01 13:03:18 153629480  2014/07/01 13:04:11
  1    17859   153629480  2014/07/01 13:04:11 153651278  2014/07/01 13:05:05
  1    17860   153651278  2014/07/01 13:05:05 153672263  2014/07/01 13:05:59

  Backup Set Copy #1 of backup set 54746
  Device Type Elapsed Time Completion Time     Compressed Tag
  ----------- ------------ ------------------- ---------- ---
  SBT_TAPE    02:52:20     2014/07/01 13:14:46 NO         TAG20140701T131434

    List of Backup Pieces for backup set 54746 Copy #1
    BP Key  Pc# Status      Media                    Piece Name
    ------- --- ----------- ---------------------- ----------
    54747   1   AVAILABLE   Oracle Recovery Appliance (ZDLRA Boston)
4qpca79s_1_1_DB1211LG

  Backup Set Copy #2 of backup set 54746
  Device Type Elapsed Time Completion Time     Compressed Tag
  ----------- ------------ ------------------- ---------- ---
  SBT_TAPE    02:52:20     2014/07/01 16:06:56 NO         TAG20140701T131434

    List of Backup Pieces for backup set 54746 Copy #2
    BP Key  Pc# Status      Media                    Piece Name
    ------- --- ----------- ---------------------- ----------
    55019   1   AVAILABLE   Oracle Recovery Appliance (ZDLRA Des Moines)
RA_SBT_54971_4qpca79s_1_2_54746_1
.
.
.
```

   In the preceding output, backup set `54746` has two copies. Copy #1 resides on `ZDLRA Boston`, which is the upstream Recovery Appliance, and copy #2 resides on `ZDLRA Des Moines`, which is the downstream Recovery Appliance.

3. With SQL*Plus or SQL Developer, connect to the *upstream* Recovery Appliance as a named user with `RASYS` abilities.

4. Confirm that the upstream Recovery Appliance has the correct replication status.

   For example, query `RA_REPLICATION_CONFIG`, which should show a state of `RUNNING` for the replication server configuration that you created in "Task 6: Create the replication server configuration on the upstream Recovery Appliance":

Copyright © 2014, 2026, Oracle and/or its affiliates.

```
SELECT REPLICATION_SERVER_NAME AS "RS_NAME",
       REPLICATION_SERVER_STATE AS "RS_STATE",
FROM   RA_REPLICATION_CONFIG;

RS_NAME         RS_STATE
--------------- --------
ZDLRADSM_REP RUNNING
```

If all preceding tests reveal the expected results, then the upstream Recovery Appliance is replicating backups of this protected database successfully.

15.6.2 Configuring Recovery Appliance Replication with TLS Using DBMS_RA

This section describes how to configure Recovery Appliance replication when TLS is in use on one or both Recovery Appliances.

Prerequisites and Assumptions

Your environment must meet the following prerequisites:

- The upstream and downstream Recovery Appliance can communicate with each other over the network.
- The downstream Recovery Appliance must be started and configured to receive backups.

The following replication cases are provided.

- Case 1: One-Way Replication; TLS disabled on Downstream
- Case 2: One-Way Replication; TLS enabled on Downstream
- Case 3: Two-Way Replication; TLS disabled on Downstream
- Case 4: Two-Way Replication; TLS enabled on Downstream
- Case 5: Two-Way Replication; TLS disabled on Upstream
- Case 6: Two-Way Replication; TLS disabled on Upstream and Downstream

15.6.2.1 Case 1: One-Way Replication; TLS disabled on Downstream

The upstream Recovery Appliance (RA1) has one-way replication to the downstream Recovery Appliance (RA2).

- The upstream Recovery Appliance can be in the mode: TLS enabled, TLS only, or TLS disabled.
- The downstream Recovery Appliance has TLS disabled.

No action is required.

15.6.2.2 Case 2: One-Way Replication; TLS enabled on Downstream

The upstream Recovery Appliance (RA1) has one-way replication to the downstream Recovery Appliance (RA2).

- The upstream Recovery Appliance (RA1) can be in the mode: TLS enabled, TLS only, or TLS disabled.
- The downstream Recovery Appliance (RA2) has TLS enabled or TLS only.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Perform these steps with RA2 as the downstream.

1. Update `tnsnames.ora` with the new TCPS information.
   a. On the downstream Recovery Appliance

      ```
      cat /u01/app/oracle/product/19.0.0.0/dbhome_1/network/admin/tnsnames.ora
      ```

   b. On the upstream Recovery Appliance, add a new entry or update the existing entry with the TCPS information from the downstream Recovery Appliance. For example:

      ```
      (ADDRESS = (PROTOCOL = TCPS)(HOST = <FULL_SCAN_NAME>)(PORT = 2484)
      ```

2. Update the trusted certificate, which has the `pem` extension, like `<NAME>.pem`
   a. Copy the trusted cert from the downstream Recovery Appliance to the upstream Recovery Appliance `tmp` directory.

      > **Note**
      >
      > Use either a different location or a different name if the upstream Recovery Appliance is TLS enabled so that the certificate on the upstream Recovery Appliance is not overwriteen.

      ```
      scp DS_RA:<trusted_cert> US_RA:/tmp/<different_name_trusted_cert>
      ```

   b. Prepare the password for the RA wallet.

      > **Note**
      >
      > Use the same password for the RA wallet and the replication wallet.

      ```
      mkstore -wrl /raacfs/raadmin/config/awallet/wallet/ -viewEntry
      oracle.security.client.password<NUMBER>
      ```

      - If the upstream Recovery Appliance is also TLS enabled, then the RA wallet already supports the certificates.

        ```
        orapki wallet add -wallet /raacfs/raadmin/config/ra_wallet/wallet
        -trusted_cert -cert /tmp/<different_name_trusted_cert>
        ```

        If the replication is bi-directional, perform the same operation but treat the local Recovery Appliance as a downstream Recovery Appliance.
      - If the upstream Recovery Appliance is not TLS enabled, then the RA wallet needs to be migrated to support the certificates.
        i. List all of the current credentials.

           ```
           mkstore –wrl /raacfs/raadmin/config/ra_wallet/wallet –
           listCredential
           ```


Copyright © 2014, 2026, Oracle and/or its affiliates.

ii. Back up the wallet.

```
mv /raacfs/raadmin/config/ra_wallet/wallet /raacfs/raadmin/
config/ra_wallet/wallet_old
```

iii. Create a new RA wallet.

```
orapki wallet create -wallet /raacfs/raadmin/config/ra_wallet/
wallet
```

iv. Import copied trusted certificate into the wallet.

```
orapki wallet add -wallet /raacfs/raadmin/config/ra_wallet/
wallet -trusted_cert
-cert /tmp/<different_name_trusted_cert>
```

v. Update wallet to auto login.

```
orapki wallet create –wallet /raacfs/raadmin/config/ra_wallet/
wallet
      -auto_login
```

vi. Recover all credentials into the new RA wallet. For each credential in the old wallet, perform:

```
mkstore –wrl /raacfs/raadmin/config/ra_wallet/wallet
-createCredential <alias> <user>  <pw>
```

c. Check to see that the replication wallet supports certificates.

```
ls -lart  /raacfs/raadmin/replication/orapki
```

This is the replication wallet standard that RACLI recommends and supports certificates.

- If the replication wallet exists, perform:

```
orapki wallet add -wallet /raacfs/raadmin/replication/orapki
-trusted_cert -cert /tmp/<different_name_trusted_cert>
```

- If the replication wallet does not exist, perform:

i. List credentials in current replication wallet.

```
mkstore –wrl /raacfs/raadmin/replication –listCredential
```

ii. Create a new replication wallet.

```
orapki wallet create -wallet /raacfs/raadmin/replication/orapki
```

iii. Import copied trusted certificate into new replication wallet.

```
orapki wallet add -wallet /raacfs/raadmin/replication/orapki
-trusted_cert --cert /tmp/<different_name_trusted_cert>
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

iv. Update wallet with auto login.

```
orapki wallet create –wallet /raacfs/raadmin/replication/orapki -
auto_login
```

v. Recover all credentials into new replication wallet

```
 mkstore –wrl /raacfs/raadmin/replication/orapki -
createCredential <tns_alias> <repl_user> <repl_user_pw>
```

3. Update the replication server parameters.

   a. Pause the replication server.

   ```
   dbms_ra.pause_replication_server()
   ```

   b. Update the replication parameters.

   - `wallet_path` should be the new replication wallet location.
   - `wallet_alias` should be the alias updated in `tnsnames.ora` in step 1

   ```
   dbms_ra.update_replication_server()
   wallet_path => 'file:/raacfs/raadmin/replication/orapki/'
   wallet_alias => 'TNS_ALIAS'
   ```

   c. Resume the replication server

   ```
   dbms_ra.resume_replication_server()
   ```

15.6.2.3 Case 3: Two-Way Replication; TLS disabled on Downstream

The upstream Recovery Appliance (RA1) has two-way replication with the downstream Recovery Appliance (RA2).

- The upstream Recovery Appliance can be in the mode: TLS enabled or TLS only.
- The downstream Recovery Appliance has TLS disabled.

Perform this step but with RA1 as the downstream. on the downstream Recovery Appliance.

1. Update `tnsnames.ora` with the new TCPS information.

   a. On the downstream Recovery Appliance

   ```
   cat /u01/app/oracle/product/19.0.0.0/dbhome_1/network/admin/tnsnames.ora
   ```

   b. On the upstream Recovery Appliance, add a new entry or update the existing entry with the TCPS information from the downstream Recovery Appliance. For example:

   ```
   (ADDRESS = (PROTOCOL = TCPS)(HOST = <FULL_SCAN_NAME>)(PORT = 2484)
   ```

2. Update the trusted certificate, which has the `pem` extension, like `<NAME>.pem`

   a. Copy the trusted cert from the downstream Recovery Appliance to the upstream Recovery Appliance `tmp` directory.

Copyright © 2014, 2026, Oracle and/or its affiliates.

> **Note**
>
> Use either a different location or a different name if the upstream Recovery Appliance is TLS enabled so that the certificate on the upstream Recovery Appliance is not overwriteen.

```
scp DS_RA:<trusted_cert> US_RA:/tmp/<different_name_trusted_cert>
```

b. Prepare the password for the RA wallet.

> **Note**
>
> Use the same password for the RA wallet and the replication wallet.

```
mkstore -wrl /raacfs/raadmin/config/awallet/wallet/ -viewEntry
oracle.security.client.password<NUMBER>
```

- If the upstream Recovery Appliance is also TLS enabled, then the RA wallet already supports the certificates.

  ```
  orapki wallet add -wallet /raacfs/raadmin/config/ra_wallet/wallet
  -trusted_cert -cert /tmp/<different_name_trusted_cert>
  ```

  If the replication is bi-directional, perform the same operation but treat the local Recovery Appliance as a downstream Recovery Appliance.

- If the upstream Recovery Appliance is not TLS enabled, then the RA wallet needs to be migrated to support the certificates.

  i. List all of the current credentials.

  ```
  mkstore –wrl /raacfs/raadmin/config/ra_wallet/wallet –
  listCredential
  ```

  ii. Back up the wallet.

  ```
  mv /raacfs/raadmin/config/ra_wallet/wallet /raacfs/raadmin/
  config/ra_wallet/wallet_old
  ```

  iii. Create a new RA wallet.

  ```
  orapki wallet create -wallet /raacfs/raadmin/config/ra_wallet/
  wallet
  ```

  iv. Import copied trusted certificate into the wallet.

  ```
  orapki wallet add -wallet /raacfs/raadmin/config/ra_wallet/
  wallet -trusted_cert
  -cert /tmp/<different_name_trusted_cert>
  ```


Copyright © 2014, 2026, Oracle and/or its affiliates.

v. Update wallet to auto login.

```
orapki wallet create –wallet /raacfs/raadmin/config/ra_wallet/
wallet
      -auto_login
```

vi. Recover all credentials into the new RA wallet. For each credential in the old wallet, perform:

```
mkstore –wrl /raacfs/raadmin/config/ra_wallet/wallet
-createCredential <alias> <user>  <pw>
```

c. Check to see that the replication wallet supports certificates.

```
ls -lart  /raacfs/raadmin/replication/orapki
```

This is the replication wallet standard that RACLI recommends and supports certificates.

- If the replication wallet exists, perform:

```
orapki wallet add -wallet /raacfs/raadmin/replication/orapki
-trusted_cert -cert /tmp/<different_name_trusted_cert>
```

- If the replication wallet does not exist, perform:

i. List credentials in current replication wallet.

```
mkstore –wrl /raacfs/raadmin/replication –listCredential
```

ii. Create a new replication wallet.

```
orapki wallet create -wallet /raacfs/raadmin/replication/orapki
```

iii. Import copied trusted certificate into new replication wallet.

```
orapki wallet add -wallet /raacfs/raadmin/replication/orapki
-trusted_cert --cert /tmp/<different_name_trusted_cert>
```

iv. Update wallet with auto login.

```
orapki wallet create –wallet /raacfs/raadmin/replication/orapki -
auto_login
```

v. Recover all credentials into new replication wallet

```
 mkstore –wrl /raacfs/raadmin/replication/orapki -
createCredential <tns_alias> <repl_user> <repl_user_pw>
```

3. Update the replication server parameters.

a. Pause the replication server.

```
dbms_ra.pause_replication_server()
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

b. Update the replication parameters.

- `wallet_path` should be the new replication wallet location.
- `wallet_alias` should be the alias updated in `tnsnames.ora` in step 1

```
dbms_ra.update_replication_server()
wallet_path => 'file:/raacfs/raadmin/replication/orapki/'
wallet_alias => 'TNS_ALIAS'
```

c. Resume the replication server

```
dbms_ra.resume_replication_server()
```

15.6.2.4 Case 4: Two-Way Replication; TLS enabled on Downstream

The upstream Recovery Appliance (RA1) has two-way replication with the downstream Recovery Appliance (RA2).

- The upstream Recovery Appliance can be in the mode: TLS enabled or TLS only.
- The downstream Recovery Appliance has TLS enabled or TLS only.

Perform this step twice: once with RA1 as the downstream, and once with RA2 as the downstream.

1. Update `tnsnames.ora` with the new TCPS information.
   a. On the downstream Recovery Appliance

   ```
   cat /u01/app/oracle/product/19.0.0.0/dbhome_1/network/admin/tnsnames.ora
   ```

   b. On the upstream Recovery Appliance, add a new entry or update the existing entry with the TCPS information from the downstream Recovery Appliance. For example:

   ```
   (ADDRESS = (PROTOCOL = TCPS)(HOST = <FULL_SCAN_NAME>)(PORT = 2484)
   ```

2. Update the trusted certificate, which has the `pem` extension, like `<NAME>.pem`
   a. Copy the trusted cert from the downstream Recovery Appliance to the upstream Recovery Appliance `tmp` directory.

   > **Note**
   >
   > Use either a different location or a different name if the upstream Recovery Appliance is TLS enabled so that the certificate on the upstream Recovery Appliance is not overwriteen.

   ```
   scp DS_RA:<trusted_cert> US_RA:/tmp/<different_name_trusted_cert>
   ```

   b. Prepare the password for the RA wallet.

Copyright © 2014, 2026, Oracle and/or its affiliates.

> **Note**
>
> Use the same password for the RA wallet and the replication wallet.

mkstore -wrl /raacfs/raadmin/config/awallet/wallet/ -viewEntry oracle.security.client.password<NUMBER>

- If the upstream Recovery Appliance is also TLS enabled, then the RA wallet already supports the certificates.

  ```
  orapki wallet add -wallet /raacfs/raadmin/config/ra_wallet/wallet
  -trusted_cert -cert /tmp/<different_name_trusted_cert>
  ```

  If the replication is bi-directional, perform the same operation but treat the local Recovery Appliance as a downstream Recovery Appliance.
- If the upstream Recovery Appliance is not TLS enabled, then the RA wallet needs to be migrated to support the certificates.

  i. List all of the current credentials.

  ```
  mkstore –wrl /raacfs/raadmin/config/ra_wallet/wallet –
  listCredential
  ```

  ii. Back up the wallet.

  ```
  mv /raacfs/raadmin/config/ra_wallet/wallet /raacfs/raadmin/
  config/ra_wallet/wallet_old
  ```

  iii. Create a new RA wallet.

  ```
  orapki wallet create -wallet /raacfs/raadmin/config/ra_wallet/
  wallet
  ```

  iv. Import copied trusted certificate into the wallet.

  ```
  orapki wallet add -wallet /raacfs/raadmin/config/ra_wallet/
  wallet -trusted_cert
  -cert /tmp/<different_name_trusted_cert>
  ```

  v. Update wallet to auto login.

  ```
  orapki wallet create –wallet /raacfs/raadmin/config/ra_wallet/
  wallet
        -auto_login
  ```

  vi. Recover all credentials into the new RA wallet. For each credential in the old wallet, perform:

  ```
  mkstore –wrl /raacfs/raadmin/config/ra_wallet/wallet
  -createCredential <alias> <user>  <pw>
  ```


Copyright © 2014, 2026, Oracle and/or its affiliates.

c. Check to see that the replication wallet supports certificates.

```
ls -lart  /raacfs/raadmin/replication/orapki
```

This is the replication wallet standard that RACLI recommends and supports certificates.

- If the replication wallet exists, perform:

```
orapki wallet add -wallet /raacfs/raadmin/replication/orapki
-trusted_cert -cert /tmp/<different_name_trusted_cert>
```

- If the replication wallet does not exist, perform:

i. List credentials in current replication wallet.

```
mkstore –wrl /raacfs/raadmin/replication –listCredential
```

ii. Create a new replication wallet.

```
orapki wallet create -wallet /raacfs/raadmin/replication/orapki
```

iii. Import copied trusted certificate into new replication wallet.

```
orapki wallet add -wallet /raacfs/raadmin/replication/orapki
-trusted_cert --cert /tmp/<different_name_trusted_cert>
```

iv. Update wallet with auto login.

```
orapki wallet create –wallet /raacfs/raadmin/replication/orapki -
auto_login
```

v. Recover all credentials into new replication wallet

```
 mkstore –wrl /raacfs/raadmin/replication/orapki -
createCredential <tns_alias> <repl_user> <repl_user_pw>
```

3. Update the replication server parameters.

a. Pause the replication server.

```
dbms_ra.pause_replication_server()
```

b. Update the replication parameters.

- `wallet_path` should be the new replication wallet location.
- `wallet_alias` should be the alias updated in `tnsnames.ora` in step 1

```
dbms_ra.update_replication_server()
wallet_path => 'file:/raacfs/raadmin/replication/orapki/’
wallet_alias => ‘TNS_ALIAS’
```

c. Resume the replication server

```
dbms_ra.resume_replication_server()
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

15.6.2.5 Case 5: Two-Way Replication; TLS disabled on Upstream

The upstream Recovery Appliance (RA1) has two-way replication with the downstream Recovery Appliance (RA2).

- The upstream Recovery Appliance has TLS disabled.
- The downstream Recovery Appliance has TLS enabled or TLS only.

Perform this step with RA2 as the downstream.

1. Update `tnsnames.ora` with the new TCPS information.
   a. On the downstream Recovery Appliance

      ```
      cat /u01/app/oracle/product/19.0.0.0/dbhome_1/network/admin/tnsnames.ora
      ```

   b. On the upstream Recovery Appliance, add a new entry or update the existing entry with the TCPS information from the downstream Recovery Appliance. For example:

      ```
      (ADDRESS = (PROTOCOL = TCPS)(HOST = <FULL_SCAN_NAME>)(PORT = 2484)
      ```

2. Update the trusted certificate, which has the `pem` extension, like `<NAME>.pem`
   a. Copy the trusted cert from the downstream Recovery Appliance to the upstream Recovery Appliance `tmp` directory.

      > **Note**
      >
      > Use either a different location or a different name if the upstream Recovery Appliance is TLS enabled so that the certificate on the upstream Recovery Appliance is not overwriteen.

      ```
      scp DS_RA:<trusted_cert> US_RA:/tmp/<different_name_trusted_cert>
      ```

   b. Prepare the password for the RA wallet.

      > **Note**
      >
      > Use the same password for the RA wallet and the replication wallet.

      ```
      mkstore -wrl /raacfs/raadmin/config/awallet/wallet/ -viewEntry
      oracle.security.client.password<NUMBER>
      ```

      - If the upstream Recovery Appliance is also TLS enabled, then the RA wallet already supports the certificates.

        ```
        orapki wallet add -wallet /raacfs/raadmin/config/ra_wallet/wallet
        -trusted_cert -cert /tmp/<different_name_trusted_cert>
        ```

        If the replication is bi-directional, perform the same operation but treat the local Recovery Appliance as a downstream Recovery Appliance.

Copyright © 2014, 2026, Oracle and/or its affiliates.

- If the upstream Recovery Appliance is not TLS enabled, then the RA wallet needs to be migrated to support the certificates.

  i. List all of the current credentials.

  ```
  mkstore –wrl /raacfs/raadmin/config/ra_wallet/wallet –
  listCredential
  ```

  ii. Back up the wallet.

  ```
  mv /raacfs/raadmin/config/ra_wallet/wallet /raacfs/raadmin/
  config/ra_wallet/wallet_old
  ```

  iii. Create a new RA wallet.

  ```
  orapki wallet create -wallet /raacfs/raadmin/config/ra_wallet/
  wallet
  ```

  iv. Import copied trusted certificate into the wallet.

  ```
  orapki wallet add -wallet /raacfs/raadmin/config/ra_wallet/
  wallet -trusted_cert
  -cert /tmp/<different_name_trusted_cert>
  ```

  v. Update wallet to auto login.

  ```
  orapki wallet create –wallet /raacfs/raadmin/config/ra_wallet/
  wallet
        -auto_login
  ```

  vi. Recover all credentials into the new RA wallet. For each credential in the old wallet, perform:

  ```
  mkstore –wrl /raacfs/raadmin/config/ra_wallet/wallet
  -createCredential <alias> <user>  <pw>
  ```

c. Check to see that the replication wallet supports certificates.

```
ls -lart  /raacfs/raadmin/replication/orapki
```

This is the replication wallet standard that RACLI recommends and supports certificates.

- If the replication wallet exists, perform:

```
orapki wallet add -wallet /raacfs/raadmin/replication/orapki
-trusted_cert -cert /tmp/<different_name_trusted_cert>
```

- If the replication wallet does not exist, perform:

  i. List credentials in current replication wallet.

  ```
  mkstore –wrl /raacfs/raadmin/replication –listCredential
  ```


Copyright © 2014, 2026, Oracle and/or its affiliates.

ii. Create a new replication wallet.

```
orapki wallet create -wallet /raacfs/raadmin/replication/orapki
```

iii. Import copied trusted certificate into new replication wallet.

```
orapki wallet add -wallet /raacfs/raadmin/replication/orapki
-trusted_cert --cert /tmp/<different_name_trusted_cert>
```

iv. Update wallet with auto login.

```
orapki wallet create –wallet /raacfs/raadmin/replication/orapki -
auto_login
```

v. Recover all credentials into new replication wallet

```
 mkstore –wrl /raacfs/raadmin/replication/orapki -
createCredential <tns_alias> <repl_user> <repl_user_pw>
```

3. Update the replication server parameters.

a. Pause the replication server.

```
dbms_ra.pause_replication_server()
```

b. Update the replication parameters.

- `wallet_path` should be the new replication wallet location.
- `wallet_alias` should be the alias updated in `tnsnames.ora` in step 1

```
dbms_ra.update_replication_server()
wallet_path => 'file:/raacfs/raadmin/replication/orapki/'
wallet_alias => 'TNS_ALIAS'
```

c. Resume the replication server

```
dbms_ra.resume_replication_server()
```

15.6.2.6 Case 6: Two-Way Replication; TLS disabled on Upstream and Downstream

The upstream Recovery Appliance (RA1) has two-way replication with the downstream Recovery Appliance (RA2).

- The upstream Recovery Appliance has TLS disabled.
- The downstream Recovery Appliance has TLS disabled.

No action is required.

15.6.3 Recovery Catalog Views for Replication

You can monitor replication using the Recovery Appliance catalog views. Table 15-3 summarizes the views that are most useful for replication.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 15-3 Views for Replication

View	Description
RA_REPLICATION_CONFIG	This view lists the replication server configurations.
RA_REPLICATION_DATABASE	This view lists information on replication servers and protected databases.
RA_REPLICATION_PAIR	This view lists replication information for replicating protection policies.
RA_REPLICATION_POLICY	This view lists the association of replication servers to protection policy.
RA_DATABASE	The `POLICY_NAME` column of this view lists the protection policy used by this protected database. The `REPLICATION_USAGE` column shows the cumulative amount of disk space (in GB) replicated for this database.
RA_PROTECTION_POLICY	This view describes the defined protection policies.

See Also

Recovery Appliance View Reference

Copyright © 2014, 2026, Oracle and/or its affiliates.

16

Implementing Additional High Availability Strategies

Besides replication, other high availability strategies can be used with Recovery Appliance to increase protection against data loss in certain scenarios.

The Oracle Maximum Availability Architecture (MAA) best practice to protect the appliance against site disasters and system outages is to implement a disaster recovery strategy using Recovery Appliance replication. With a replica appliance, protected database backup, redo, and restore operations continue uninterrupted, preserving complete data protection.

If your organization does not have a disaster recovery strategy or if you would like to add local system high availability to your existing disaster recovery strategy, you can use the Backup and Redo Failover feature of Recovery Appliance.

Another component of a high availability (HA) and disaster recovery solution is Oracle Data Guard. Oracle Data Guard minimizes service interruption and resulting data loss by maintaining a synchronized standby database for the protected database.

See Also

- "Managing Temporary Outages with a Backup and Redo Failover Strategy" for information and instructions for configuring Backup and Redo Failover
- "Maximum Availability: Recovery Appliance with Oracle Data Guard" for information about Oracle Data Guard
- "Replicating Backups with Recovery Appliance " for information about Recovery Appliance replication

16.1 Managing Temporary Outages with a Backup and Redo Failover Strategy

Backup and Redo Failover is a high availability feature that allows protected databases to temporarily direct backups and redo to an alternate Recovery Appliance when the primary Recovery Appliance experiences an outage or requires planned maintenance. This allows protected database backups and redo to continue uninterrupted and preserves complete data protection. It also prevents the local archived log destinations of the database from filling up and impacting the database, which can occur with no alternate backup destination.

16.1.1 Overview of the Backup and Redo Failover Feature

In an environment where Backup and Redo Failover is configured, a protected database sends backups and redo to a primary Recovery Appliance under normal circumstances. When that appliance is unavailable, the protected database sends backups and redo to an alternate Recovery Appliance until service on the primary is restored.

Copyright © 2014, 2026, Oracle and/or its affiliates.

The alternate appliance does not create virtual full backups from the temporary backups it receives; it only stores the backup pieces (incremental and archived log backups). When the primary appliance is back online and operational, the alternate appliance forwards all temporary backups to the primary appliance, which uses them to create the corresponding virtual full backups. After all virtual full backups are created, the protected database resumes sending backups and redo to the primary appliance. The alternate appliance deletes the temporary backup pieces from local storage only after they are successfully forwarded to the primary appliance.

16.1.2 Configuring Backup and Redo Failover

This section explains how to configure Backup and Redo Failover. The basic work flow is as follows:

1. Configure the primary Recovery Appliance, as described in "Configuring the Primary Recovery Appliance for Backup and Redo Failover".
2. Configure the alternate Recovery Appliance, as described in "Configuring the Alternate Recovery Appliance for Backup and Redo Failover".
3. Configure replication from the alternate Recovery Appliance to the primary Recovery Appliance, as described in "Configuring Replication for Backup and Redo Failover".
4. Configure the protected database to send backups, as described in "Configuring the Protected Database for Backup and Redo Failover".

16.1.2.1 Configuring the Primary Recovery Appliance for Backup and Redo Failover

To configure the primary Recovery Appliance for Backup and Redo Failover, you perform many of the tasks for setting up a downstream Recovery Appliance in a replication scenario.

Task 1: Create a VPC user account and a replication user account on the primary Recovery Appliance

Follow the instructions in "racli add db_user".

For example, log in to the Recovery appliance as root, change to the bin directory, and use the following command to create the VPC user:

```
# ./racli add db_user --user_name=vpcuser --user_type=vpc
```

Enter the password for `vpcuser` user when prompted.

To create the replication user `repuser_from_alternate` with the `CREATE SESSION` privilege:

```
CREATE USER repuser_from_alternate IDENTIFIED BY password;
GRANT CREATE SESSION TO repuser_from_alternate;
```

The `user_name` created on the alternate must be the same as the VPC user created on the primary. However, the passwords do not need to be the same.

Task 2 Create a protection policy on the primary Recovery Appliance

Follow the instructions in Creating a Protection Policy. Ensure that the `store_and_forward` field is set to `NO`.

For example, execute the following PL/SQL program to create a `primary_brf` policy:

Copyright © 2014, 2026, Oracle and/or its affiliates.

```
BEGIN
  DBMS_RA.CREATE_PROTECTION_POLICY (
    protection_policy_name => 'primary_brf',
    description            => 'For protected dbs on primary',
    storage_location_name  => 'delta',
    recovery_window_goal   => INTERVAL '28' DAY,
    guaranteed_copy        => 'NO',
    store_and_forward      => 'NO');
END;
```

Task 3: Add a database to the protection policy on the primary Recovery Appliance

Follow the instructions in "Enrolling Protected Databases".

For example, execute the following PL/SQL program to add `orcl12` to the `primary_brf` policy that you created in the previous task:

```
BEGIN
  DBMS_RA.ADD_DB (
    db_unique_name         => 'orcl12',
    protection_policy_name => 'primary_brf',
    reserved_space         => '128G');
END;
```

Task 4: Grant database access to the VPC user and the replication user on the primary Recovery Appliance

Follow the instructions in "Enrolling Protected Databases".

For example, execute the following PL/SQL programs to grant the VPC user `vpcuser` and the replication user `repuser_from_alternate` the required privileges on protected database `orcl12`:

```
BEGIN
  DBMS_RA.GRANT_DB_ACCESS (
    username       => 'vpcuser',
    db_unique_name => 'orcl12');
END;
BEGIN
  DBMS_RA.GRANT_DB_ACCESS (
    username       => 'repuser_from_alternate',
    db_unique_name => 'orcl12');
END;
```

16.1.2.2 Configuring the Alternate Recovery Appliance for Backup and Redo Failover

To configure the alternate Recovery Appliance, you perform tasks similar to setting up an upstream Recovery Appliance in a replication scenario.

Task 1 Create a protection policy for Backup and Redo Failover on the alternate Recovery Appliance

Follow the instructions in Creating a Protection Policy. Ensure that you set the `store_and_forward` field to `YES`.

For example, execute the following PL/SQL program to create an `alt_brf` policy:

```
BEGIN
  DBMS_RA.CREATE_PROTECTION_POLICY (
    protection_policy_name => 'alt_brf',
    description            => 'For protected dbs on alternate',
    storage_location_name  => 'delta',
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
    recovery_window_goal   => INTERVAL '28' DAY,
    guaranteed_copy        => 'NO',
    store_and_forward      => 'YES');
END;
```

Task 2: Add the database to the protection policy on the alternate Recovery Appliance

Follow the instructions in Enrolling Protected Databases.

For example, execute the following PL/SQL program to add `orcl12` to the `alt_brf` policy that you created in the previous task:

```
BEGIN
  DBMS_RA.ADD_DB (
    db_unique_name         => 'orcl12',
    protection_policy_name => 'alt_brf',
    reserved_space         => '128G');
END;
```

Task 3: Grant database access to the VPC user on the alternate Recovery Appliance

You created this user in "Task 1: Create a VPC user account and a replication user account on the primary Recovery Appliance".

Follow the instructions in Enrolling Protected Databases.

For example, execute the following PL/SQL program to grant the VPC user `vpcuser` the required privileges on protected database `orcl12` :

```
BEGIN
  DBMS_RA.GRANT_DB_ACCESS (
    username       => 'vpcuser',
    db_unique_name => 'orcl12');
END;
```

16.1.2.3 Configuring Replication for Backup and Redo Failover

After you configure the primary and the alternate Recovery Appliances, you perform tasks similar to setting up replication from the alternate to the primary appliance. In this scenario, the alternate appliance has the upstream role and the primary appliance has the downstream role.

Task 1: Configure an Oracle wallet on the alternate Recovery Appliance

On the alternate Recovery Appliance, use the `mkstore` utility to create an Oracle auto-login wallet and add the credentials for the replication user you created in "Task 1: Create a VPC user account and a replication user account on the primary Recovery Appliance". The alternate Recovery Appliance requires these credentials when it logs in to the primary Recovery Appliance.

To configure an auto-login wallet on the alternate Recovery Appliance:

1. Run the following command to create an Oracle wallet in the `/dbfs_repdbfs/REPLICATION` directory:

   ```
   mkstore -wrl /dbfs_repdbfs/REPLICATION -createALO
   ```

   The `mkstore` utility creates a file named `cwallet.sso` in the designated location.

2. Run the following command to add the replication user credentials:

   ```
   mkstore -wrl wallet_location -createCredential serv_name rep_user pwd
   ```


Copyright © 2014, 2026, Oracle and/or its affiliates.

The placeholders are defined as follows:

- *wallet_location* is the directory in which you created the wallet in the previous step.
- *serv_name* is the Oracle network service name that you use in an EZ Connect descriptor to identify the primary Recovery Appliance on the Oracle network.
- *rep_user* is the user name of the replication user account. This user was created in Task 1: Create a VPC user account and a replication user account on the primary Recovery Appliance . The replication user is not created on the alternate.
- *pwd* is the secure password of the replication user *rep_user*.

For example, the following command adds credentials for the net service name `rapribrf-scan.acme.com` using port `1522` and a database name of `rapri`, and the replication user name `repuser_from_alternate`:

```
mkstore -wrl /dbfs_repdbfs/REPLICATION -createCredential \
"rapribrf-scan.acme.com:1522/rapri" "repuser_from_alternate" "pwd"
```

3. Verify that the credentials were properly added for this user by running the following command:

```
mkstore -wrl /dbfs_repdbfs/REPLICATION -listCredential

Oracle Secret Store Tool : Version 12.1.0.1 Copyright (c) 2004, 2012, Oracle and/or
its affiliates. All rights reserved.
List credential (index: connect_string username)
1: rapribrf-scan.acme.com:1522/rapri repuser_from_alternate
```

The results do not display the password.

Task 2: Create the replication server configuration on the alternate Recovery Appliance

For the primary Recovery Appliance to which this alternate Recovery Appliance will forward backups after an outage, create a replication server configuration by executing `DBMS_RA.CREATE_REPLICATION_SERVER`.

This task assumes the following:

- You want to create a replication server configuration named `raprimary_rep`.
- You want the alternate Recovery Appliance to log in to the primary Recovery Appliance using the replication account `repuser_from_alternate`. You created this account in "Task 1: Create a VPC user account and a replication user account on the primary Recovery Appliance".
- The configuration uses the net service name `rapribrf-scan.acme.com:1522/rapri` that you stored in the Oracle wallet you created in "Task 1: Configure an Oracle wallet on the alternate Recovery Appliance".
- The Oracle wallet is stored in `/dbfs_repdbfs/REPLICATION`.
- The file name of the Recovery Appliance Backup Module, which is preinstalled on every Recovery Appliance, is `/u01/app/oracle/product/12.1.0.2/dbh1/lib/libra.so`. The module functions as an SBT media management library. RMAN references this module when allocating or configuring a channel for backup to the Recovery Appliance (see "Configuring a Protected Database for Recovery Appliance Replication").

To create the replication server configuration:

1. With SQL*Plus or SQL Developer, connect to the alternate Recovery Appliance metadata database as `RASYS`.

Copyright © 2014, 2026, Oracle and/or its affiliates.

2. Run the `DBMS_RA.CREATE_REPLICATION_SERVER` procedure for the primary Recovery Appliance.

   The following example creates the replication server configuration named `raprimary_rep` for the primary Recovery Appliance:

```
BEGIN
  DBMS_RA.CREATE_REPLICATION_SERVER (
    replication_server_name => 'raprimary_rep',
    sbt_so_name      => '/u01/app/oracle/product/12.1.0.2/dbh1/lib/libra.so',
    catalog_user_name       => 'RASYS',
    wallet_alias            => 'rapribrf-scan.acme.com:1522/rapri',
    wallet_path             => 'file:/dbfs_repdbfs/REPLICATION');
END;
```

3. Confirm the creation of the replication server configuration. The `replication_server_name` is converted to upper-case and stored as such. Therefore queries with the name should also be upper-case.

   For example, run the following query:

```
SELECT COUNT(*) should_be_one
FROM   RA_REPLICATION_SERVER
WHERE  REPLICATION_SERVER_NAME = 'RAPRIMARY_REP';

SHOULD_BE_ONE
-------------
1
```

   If the configuration was created correctly, then the return value is `1`.

Task 3: Associate the alternate Recovery Appliance with the protection policy for Backup and Redo Failover

Specify the primary Recovery Appliance to which the alternate Recovery Appliance forwards backups after an outage by assigning the replication server configuration to a protection policy.

This task assumes the following:

- You want to use the replication server configuration named `raprimary_rep`, which you created in "Task 2: Create the replication server configuration on the alternate Recovery Appliance".
- You want to add the replication server configuration to protection policy `alt_brf`, which you created in "Task 1 Create a protection policy for Backup and Redo Failover on the alternate Recovery Appliance".

To associate the replication server configuration with the Backup and Redo Failover protection policy:

1. Ensure you are connected to the metadata database on the alternate Recovery Appliance as the Recovery Appliance administrator.
2. Run the `DBMS_RA.ADD_REPLICATION_SERVER` procedure for the Backup and Redo Failover protection policy and replication server configuration.

   For example, execute the following PL/SQL program:

```
BEGIN
  DBMS_RA.ADD_REPLICATION_SERVER (
   replication_server_name => 'raprimary_rep',
   protection_policy_name  => 'alt_brf');
END;
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

> **See Also**
>
> "ADD_REPLICATION_SERVER"

16.1.2.4 Configuring the Protected Database for Backup and Redo Failover

After you configure replication for Backup and Redo Failover, the protected database administrator should perform the tasks in this section so that the protected database can send backups to the primary Recovery Appliance under normal conditions, and to the alternate Recovery Appliance during a planned or unplanned outage.

Task 1: Configure sqlnet.ora to point to the wallet location

Ensure that the sqlnet.ora file contains the location of the Oracle wallet.

The following example shows how the wallet location entry should appear:

```
SQLNET.WALLET_OVERRIDE = true
WALLET_LOCATION =
(SOURCE =
    (METHOD = FILE)
    (METHOD_DATA =
      (DIRECTORY = /u01/app/oracle/product/12.1.0/dbhome_1/dbs/zdlra)
   )
  )
```

Task 2: Create an auto-login wallet in the location specified in sqlnet.ora

The following example creates an auto-login wallet in the directory specified in "Task 1: Configure sqlnet.ora to point to the wallet location":

```
$ mkstore -wrl /u01/app/oracle/product/12.1.0/dbhome_1/dbs/zdlra/ -createALO
```

Task 3: Add the credentials for the primary and alternate Recovery Appliances to the wallet

In this task the protected database administrator adds credentials for the primary and alternate appliances using the VPC user you created in "Task 1: Create a VPC user account and a replication user account on the primary Recovery Appliance" to the wallet.

The following examples add the `vpcuser` credentials for the primary appliance `rapribrf-scan.acme.com:1521/rapri:dedicated` and the alternate appliance `raaltbrf-scan.acme.com:1521/raalt:dedicated` to the wallet on the protected database:

```
$ mkstore -wrl /u01/app/oracle/product/12.1.0/dbhome_1/dbs/zdlra/ -createCredential
"rapribrf-scan.acme.com:1521/rapri:dedicated" "vpcuser" "pwd"
$ mkstore -wrl /u01/app/oracle/product/12.1.0/dbhome_1/dbs/zdlra/ -createCredential
"raaltbrf-scan.acme.com:1521/raalt:dedicated" "vpcuser" "pwd"
```

Task 4: Register the database with the alternate Recovery Appliance and back up the control file

For this task, the protected database administrator performs steps 1 and 2, and the Recovery Appliance administrator performs step 3.

To register the database and back up the control file:

1. Using RMAN, connect to the protected database as `TARGET` and to the alternate Recovery Appliance catalog as `CATALOG`, and then run the `REGISTER DATABASE` command.

Copyright © 2014, 2026, Oracle and/or its affiliates.

2. After the `REGISTER DATABASE` command is completed, back up the current control file to the alternate appliance:

   ```
   BACKUP DEVICE TYPE SBT CURRENT CONTROLFILE;
   ```

3. Verify that the control file backup was replicated from the alternate appliance to the primary appliance.

Task 5: Ensure that the database is registered with the primary Recovery Appliance

This step is to confirm that the protected database is registered with the primary appliance. Because replication is configured when the database is registered with the alternate appliance in the previous task, the database should automatically be registered with the primary appliance.

To confirm registration with the primary appliance:

1. In RMAN, connect to the database using the primary appliance credentials in the CATALOG connect string.

   ```
   rman TARGET / CATALOG /@rapribrf-scan.acme.com:1521/rapri:dedicated
   ```

2. Run the `REGISTER DATABASE` command.

   The following error should display:

   ```
   RMAN-20002: target database already registered in recovery catalog
   ```

> **Note**
>
> - The protected database administrator must also create a separate `RMAN` backup script that directs backups to the alternate Recovery Appliance when the primary appliance is not available, and redirects backups to the primary appliance when it is back in service. This script must connect to the alternate Recovery Appliance catalog and have the `CONFIGURE CHANNEL` or `ALLOCATE CHANNEL` command with `credential_alias` set to the alternate appliance. See *Zero Data Loss Recovery Appliance Protected Database Configuration Guide* for an example of how to create an RMAN backup script for the Recovery Appliance.
> - To send real time redo to the alternate Recovery Appliance during the outage of the primary appliance, an additional log archive destination must be defined as an `ALTERNATE` for the log archive destination used to connect to the primary appliance. The connect string must be defined in the Oracle auto-login wallet, similar to the connect string required for the primary appliance, and using the same VPC user (although the password may be different). See *Data Guard Concepts and Administration* for an example of how to use the `ALTERNATE` attribute to automatically fail over to a alternate remote destination.

16.2 Implementing DR Failover to Downstream Recovery Appliance

This section provides steps on how to configure a protected database for transparent failover of backup operations and redo transport to a downstream Recovery Appliance.

As part of disaster recovery, protected databases should failover to a downstream Recovery Appliance as the target for sending backup files and redo transport if the upstream Recovery Appliance is unavailable.

Copyright © 2014, 2026, Oracle and/or its affiliates.

For sake of clarity, this examples makes the following assumptions:

- If you have real time redo transport enabled, it receives an error and stops sending redo to the upstream Recovery Appliance. Within a minute, real time redo transport connects to the downstream Recovery Appliance and resumes sending redo there.
- The name of the example protected database is `CDB122DR`. It is a Container Databases with One Pluggable Database.
- The name of the example upstream Recovery Appliance is `RAHADR1`.
- The name of the example downstream Recovery Appliance is `RAHADR2`.
- A common VPC user called `HADR_COMMON_VPCUSER` was created on both Recovery Appliances and ***must*** use the same password on both.
- A local VPC user called `HADR_LOCAL_VPCUSER` has been created on both Recovery Appliances but the password can be different between the two.
- The replication server between `RAHADR1` and `RAHADR2` is using the VPC user `REPUSER_FROM_HADR1`.

When using a Data Guard setup that has primary and standby databases, they have the same `dbid` and `dbname`, so each must have a different `db_unique_name`. Use a unique control file `autobackup` format at primary and standby to guarantee uniqueness. The format can be specified by using RMAN configuration settings. Default `Controlfile Autobackup Format`:

```
CONFIGURE CONTROLFILE AUTOBACKUP FORMAT FOR DEVICE TYPE SBT_TAPE TO '%F';
```

Add `db_unique_name` to the default format for both primary and standby databases:

```
CONFIGURE CONTROLFILE AUTOBACKUP FORMAT FOR DEVICE TYPE SBT_TAPE TO
'<db_unique_name>_%F';
```

The `db_unique_name` is obtained from from `v$database`.

```
select db_unique_name from v$database;
```

16.2.1 Setup and Configuration for Failover

This section establishes VPC users for the Recovery Appliances to use later for failover.It modifies the network configuration files needed, configures the replication server, creates protection policies, registers the protected database, and adds several grants to the upstream and downstream Recovery Appliances.

16.2.1.1 Creating VPC Users

This task creates database VPC user accounts in the upstream and downstream Recovery Appliances.

When creating the accounts, keep in mind these password requirements.

- The first VPC user (`HADR_LOCAL_VPCUSER`) account may be used by other protected databases and can have different passwords between the RAHADR1 and RAHADR2 Recovery Appliances.

Copyright © 2014, 2026, Oracle and/or its affiliates.

- The second VPC user (`HADR_COMMON_VPCUSER`) account must use the same password on both the RAHADR1 and RAHADR2 Recovery Appliances and can be used by other protected databases

The following conditions are applicable to this specific example.

- Recovery Appliance RAHADR1 has previously been installed with a `DB_UNIQUE_NAME` of `rahadr1`.
- Recovery Appliance RAHADR2 has previously been installed with a `DB_UNIQUE_NAME` of `rahadr2`.

1. Create two VPC users for the protected database on the upstream Recovery Appliance RAHADR1.

```
# racli add db_user --user_name HADR_LOCAL_VPCUSER --user_type=vpc
[HADR_LOCAL_VPCUSER] New Password: ***********
Sun Mar 25 08:27:53 2018: Start: Add vpc user HADR_LOCAL_VPCUSER.
Sun Mar 25 08:27:53 2018: Add vpc user HADR_LOCAL_VPCUSER successfully.
Sun Mar 25 08:27:53 2018: End: Add vpc user HADR_LOCAL_VPCUSER.

# racli add db_user --user_name HADR_COMMON_VPCUSER --user_type=vpc
[HADR_COMMON_VPCUSER] New Password: ***********
Sun Mar 25 08:27:53 2018: Start: Add vpc user HADR_COMMON_VPCUSER.
Sun Mar 25 08:27:53 2018: Add vpc user HADR_COMMON_VPCUSER successfully.
Sun Mar 25 08:27:53 2018: End: Add vpc user HADR_COMMON_VPCUSER.
```

2. Create two VPC users for the protected database on the downstream Recovery Appliance RAHADR2.

```
# racli add db_user --user_name HADR_LOCAL_VPCUSER --user_type=vpc
[HADR_LOCAL_VPCUSER] New Password: ***********
Sun Mar 25 08:27:53 2018: Start: Add vpc user HADR_LOCAL_VPCUSER.
Sun Mar 25 08:27:53 2018: Add vpc user HADR_LOCAL_VPCUSER successfully.
Sun Mar 25 08:27:53 2018: End: Add vpc user HADR_LOCAL_VPCUSER.

# racli add db_user --user_name HADR_COMMON_VPCUSER --user_type=vpc
[HADR_COMMON_VPCUSER] New Password: ***********
Sun Mar 25 08:27:53 2018: Start: Add vpc user HADR_COMMON_VPCUSER.
Sun Mar 25 08:27:53 2018: Add vpc user HADR_COMMON_VPCUSER successfully.
Sun Mar 25 08:27:53 2018: End: Add vpc user HADR_COMMON_VPCUSER.
```

3. If the VPC user account used by the replication server for sending backups from the upstream (HARADR1) to the downstream (RAHADR2) Recovery Appliances hasn't been created, create the VPC user now.

```
# racli add db_user --user_name REPUSER_FROM_HADR1 --user_type=vpc
[REPUSER_FROM_HADR1] New Password: ***********

Sun Mar 25 08:35:01 2018: Start: Add vpc user REPUSER_FROM_HADR1.
Sun Mar 25 08:35:01 2018: Add vpc user REPUSER_FROM_HADR1 successfully.
Sun Mar 25 08:35:01 2018: End: Add vpc user REPUSER_FROM_HADR1.
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

16.2.1.2 Modifying Configuration for Transport Failover

This task modifies the Oracle network configuration files that are used for transparent failover to the downstream Recovery Appliance.

If you have a RAC database, this should be performed on each host where the protected database runs.

1. Verify that there are no `${ORACLE_HOME}/dbs/ra${ORACLE_SID}.ora` files on any of the hosts.

   This file has the effect of overriding all the configuration parameters defined in this step and should be removed if present.

2. Configure a TNS alias in the `tnsnames.ora` file that will be used by RMAN to connect to the correct Recovery Appliance.

   ```
   $ cd ${ORACLE_HOME}/network/admin
   ```

3. Edit `tnsnames.ora` and add the following entry:

   ```
   DR_RAHADR =
   (DESCRIPTION_LIST =
     (LOAD_BALANCE = off)
     (FAILOVER = on)
     (DESCRIPTION =
       (CONNECT_TIMEOUT = 5)
       (TRANSPORT_CONNECT_TIMEOUT = 3)
       (RETRY_COUNT = 3)
       (ADDRESS_LIST =
         (ADDRESS = (PROTOCOL = TCP)(HOST = ra1ingest-scan)(PORT = 1521))
       )
       (CONNECT_DATA =
         (SERVICE_NAME = rahadr1)
       )
     )
     (DESCRIPTION =
       (CONNECT_TIMEOUT = 5)
       (TRANSPORT_CONNECT_TIMEOUT = 3)
       (RETRY_COUNT = 3)
       (ADDRESS_LIST =
         (ADDRESS = (PROTOCOL = TCP)(HOST = ra2ingest-scan)(PORT = 1521))
       )
       (CONNECT_DATA =
         (SERVICE_NAME = rahadr2)
       )
     )
   )
   DR_RAHADR1 =
   (DESCRIPTION_LIST =
     (DESCRIPTION =
       (CONNECT_TIMEOUT = 5)
       (TRANSPORT_CONNECT_TIMEOUT = 3)
       (RETRY_COUNT = 3)
       (ADDRESS_LIST =
         (ADDRESS = (PROTOCOL = TCP)(HOST = ra1ingest-scan)(PORT = 1521))
   ```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
    )
    (CONNECT_DATA =
      (SERVICE_NAME = rahadr1)
    )
  )
)
DR_RAHADR2 =
(DESCRIPTION_LIST =
  (DESCRIPTION =
    (CONNECT_TIMEOUT = 5)
    (TRANSPORT_CONNECT_TIMEOUT = 3)
    (RETRY_COUNT = 3)
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP)(HOST = ra2ingest-scan)(PORT = 1521))
    )
    (CONNECT_DATA =
      (SERVICE_NAME = rahadr2)
    )
  )
)
```

16.2.1.3 Configuring the Replication Server

This task configures the replication server that sends the database backups from RAHADR1 to the RAHADR2 Recovery Appliance.

The operations and naming conventions used here are equivalent to those in Enterprise Manager when there is no dedicated replication network. For more information, see Replicating Backups with Recovery Appliance .

The replication server between RAHADR1 and RAHADR2 has not already been created.

1. If a replication wallet does not exist on RAHADR1, create a replication wallet that points to RAHADR2.

   ```
   $ mkstore -wrl file:/dbfs_repdbfs/REPLICATION -createALO
   ```

2. Add the credentials to the wallet. On RAHADR1, add the credentials for logging into RAHADR2.

   ```
   $ mkstore -wrl file:/dbfs_repdbfs/REPLICATION
   -createCredential <rahadr2-scan>:1521/rahadr2 REPUSER_FROM_HADR1
   my_v3ry_c0mplex_pa55w0rd
   ```

3. Create the Recovery Appliance replication server on RAHADR1.

   ```
   $ sqlplus rasys/ra

   SQL> exec dbms_ra.create_replication_server(
   replication_server_name =>'RAHADR2_REP',
   sbt_so_name => 'libra.so', max_streams => 8,
   catalog_user_name=> 'RASYS',
   wallet_alias => '<rahadr2-scan>:1521/rahadr2',
   wallet_path => 'file:/dbfs_repdbfs/REPLICATION');

   PL/SQL procedure successfully completed.
   ```


Copyright © 2014, 2026, Oracle and/or its affiliates.

16.2.1.4 Configuring Upstream and Downstream Recovery Appliances

This task configures the protection policies for the protected database on the downstream and upstream Recovery Appliance, and then adds the protection policy to the replication server.

If a protection policy that is used, for example, by the CBR122DR database does not exist on the respective Recovery Appliances, these steps create them. The protection policy name does not have to be unique between the downstream and upstream Recovery Appliances.

To prevent a circular references between RAHADR1 and RAHADR2, the protection policy from RAHADR2 is not added to the replication server while the protection policy from RAHADR1 is added. All databases in the protection policy are replicated.

Note: Because RAHADR2 does not normally accept redo from the CDB122DR database, set the unprotected data window parameter is set to 1.25 days to avoid false alerts from occurring if the CDB122DR database is idle.

1. Log into SQLPLS as `rasys/ra` on RAHADR2, the downstream Recovery Appliance. This step and the next few are performed on RAHADR2, unless otherwise stated.

   ```
   $ sqlplus rasys/ra
   ```

2. Create the protection policy.

   ```
   SQL> exec dbms_ra.create_protection_policy(
   protection_policy_name => 'cdb122dr_PP',
   storage_location_name => 'DELTA',
   recovery_window_goal => numtodsinterval(3,'DAY'),
   unprotected_window => numtodsinterval(1.25,'DAY'),
   allow_backup_deletion => 'NO');

   PL/SQL procedure successfully completed.
   ```

3. Add the database (for this example) and its protection policy to the list of those protected by the Recovery Appliance.

   ```
   SQL> exec dbms_ra.add_db(
   db_unique_name => 'cdb122dr',
   protection_policy_name=> 'cdb122dr_PP',
   reserved_space => '1T');

   PL/SQL procedure successfully completed.
   ```

4. Grant access to the replication user to the database (for this example).

   ```
   SQL> exec dbms_ra.grant_db_access(
   username => 'REPUSER_FROM_HADR1',
   db_unique_name => 'cdb122dr');

   PL/SQL procedure successfully completed.
   ```

5. Log into sqlplus as `rasys/ra` on RAHADR1, the upstream Recovery Appliance. This step and all that follow are performed on RAHADR1.

   ```
   $ sqlplus rasys/ra
   ```


Copyright © 2014, 2026, Oracle and/or its affiliates.

6. Create a protection policy. The protection policy name does not have to be unique.

```
SQL> exec dbms_ra.create_protection_policy(
protection_policy_name =>'cdb122dr_PP',
storage_location_name => 'DELTA',
recovery_window_goal => numtodsinterval(3,'DAY'),
unprotected_window => numtodsinterval(5,'MINUTE'),
allow_backup_deletion => 'NO');

PL/SQL procedure successfully completed.
```

7. Add the database (for this example) and its protection policy to the list of those protected by the Recovery Appliance.

```
SQL> exec dbms_ra.add_db(
db_unique_name => 'cdb122dr',
protection_policy_name=> 'cdb122dr_PP',
reserved_space => '1T');

PL/SQL procedure successfully completed.
```

8. Grant access to the replication user to the database (for this example).

```
SQL> exec dbms_ra.grant_db_access(
username => 'HADR_LOCAL_VPCUSER',
db_unique_name => 'cdb122dr');

PL/SQL procedure successfully completed.
```

9. Add the protection policy to the replication server. This step is performed on the upstream Recovery Appliance (RAHADR1). This step was not performed on the downstream Recovery Appliance, in order to prevent a circular reference between the two Recovery Appliances.

```
SQL> exec dbms_ra.add_replication_server(
replication_server_name =>'RAHADR2_REP',
protection_policy_name => 'cdb122dr_PP');

PL/SQL procedure successfully completed.
```

16.2.1.5 Registering the Protected Database on the Upstream Recovery Appliance

This task configures the wallet, adds VPC user credentials, tests those credentials, and registers the protected database with the upstream Recovery Appliance. If it is a RAC database, the steps need to be performed on each host where the protected database runs.

The operations and naming conventions used here are equivalent to those in Enterprise Manager.

1. Configure the `sqlnet.ora` file that will be used by RMAN to connect to the correctRecovery Appliance. Go to the proper directory.

```
$ cd ${ORACLE_HOME}/network/admin
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

2. Edit the `sqlnet.ora` file and ensure the following parameters are set correctly:

```
SQLNET.WALLET_OVERRIDE = true

NAMES.DIRECTORY_PATH= (TNSNAMES, EZCONNECT)

WALLET_LOCATION =
  (SOURCE =
    (METHOD = FILE)
    (METHOD_DATA =
      (DIRECTORY = /u01/app/oracle/product/12.2.0.1/dbhome_1/dbs/zdlra)
    )
  )

SQLNET.EXPIRE_TIME = 10
```

3. Create a replication wallet that stores each of the VPC user credentials. Perform this step only if the replication wallet doesn't already exist. On each host:

```
$ mkstore -wrl file:/u01/app/oracle/product/12.2.0.1/dbhome_1/dbs/zdlra -
createALO
```

4. Create credential aliases for each of the three credentials that will be used by RMAN. On each host, run the `mkstore` command. Enter the appropriate password when prompted.

```
$ mkstore -wrl file:/u01/app/oracle/product/12.2.0.1/dbhome_1/dbs/zdlra -
createCredential dr_rahadr2 hadr_local_vpcuser hadr2_L0cal_Pa55w0rd

$ mkstore -wrl file:/u01/app/oracle/product/12.2.0.1/dbhome_1/dbs/zdlra -
createCredential dr_rahadr1 hadr_local_vpcuser hadr1_L0cal_Pa55w0rd

$ mkstore -wrl file:/u01/app/oracle/product/12.2.0.1/dbhome_1/dbs/zdlra -
createCredential dr_rahadr hadr_common_vpcuser c0mm0n_Pa55w0rd
```

5. Verify the credentials are working correctly by logging into each target using only the credential alias. On each host, run the following:

```
$ sqlplus /@dr_rahadr1
```

6. Register the protected database with the Recovery Appliance in RAHADR1. On one of the hosts, run:

```
$ rman target / catalog /@dr_rahadr1

RMAN> register database;
```

7. Perform a test backup of the current control file to Recovery Appliance hadr1 (RAHADR1). On one of the protected database hosts, perform a backup of the current control file.

```
$ rman target / catalog /@dr_rahadr1

RMAN> CONFIGURE CHANNEL DEVICE TYPE 'SBT_TAPE' FORMAT '%d_%U' PARMS
"SBT_LIBRARY=/u01/app/oracle/product/12.2.0.1/dbhome_1/lib/libra.so,
ENV=(RA_WALLET='location=file:/u01/app/oracle/product/12.2.0.1/
dbhome_1/dbs/z
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
dlra credential_alias=dr_rahadr1')";

RMAN> backup device type sbt current controlfile tag 'controltest';

Starting backup at 05-JUN-18
allocated channel: ORA_SBT_TAPE_1
channel ORA_SBT_TAPE_1: SID=2320 instance=cdb122dr1 device type=SBT_TAPE
channel ORA_SBT_TAPE_1: RA Library (RAHADR1)
SID=6DE9FE3D49ED4598E05311F3850AC59F
allocated channel: ORA_SBT_TAPE_2
channel ORA_SBT_TAPE_2: SID=2516 instance=cdb122dr1 device type=SBT_TAPE
channel ORA_SBT_TAPE_2: RA Library (RAHADR1)
SID=6DE9FE48D84C48C8E05311F3850A89BE
channel ORA_SBT_TAPE_1: starting full datafile backup set
channel ORA_SBT_TAPE_1: specifying datafile(s) in backup set
including current control file in backup set
channel ORA_SBT_TAPE_1: starting piece 1 at 05-JUN-18
channel ORA_SBT_TAPE_1: finished piece 1 at 05-JUN-18
piece handle=CDB122DR_2kt4m80u_1_1 tag=CONTROLTEST comment=API Version
2.0,MMS Version 3.17.1.26
channel ORA_SBT_TAPE_1: backup set complete, elapsed time: 00:00:15
Finished backup at 05-JUN-18
Starting Control File and SPFILE Autobackup at 05-JUN-18
piece handle=c-3244939197-20180605-00 comment=API Version 2.0,MMS Version
3.17.1.26
Finished Control File and SPFILE Autobackup at 05-JUN-18
```

8. List the backup set just created. Verify there are two copies of the control file, one on Recovery Appliance hadr1 (RAHADR1) and the other on Recovery Appliance hadr2 (RAHADR2).

```
RMAN> list backupset tag CONTROLTEST;

List of Backup Sets
===================
BS Key Type LV Size
------- ---- -- ----------
220 Full 138.75M
Control File Included: Ckp SCN: 9076177 Ckp time: 05-JUN-18
Backup Set Copy #1 of backup set 220
Device Type Elapsed Time Completion Time Compressed Tag
---------- ------------ --------------- ---------- ---
SBT_TAPE 07:00:21 05-JUN-18 NO CONTROLTEST
List of Backup Pieces for backup set 220 Copy #1
BP Key Pc# Status Media Piece Name
------- --- ----------- ---------------------- ----------
221 1 AVAILABLE Recovery Appliance (RAHADR1)
CDB122DR_2kt4m80u_1_1
Backup Set Copy #2 of backup set 220
Device Type Elapsed Time Completion Time Compressed Tag
---------- ------------ --------------- ---------- ---
SBT_TAPE 07:00:21 05-JUN-18 NO CONTROLTEST
List of Backup Pieces for backup set 220 Copy #2
BP Key Pc# Status Media Piece Name
------- --- ----------- ---------------------- ----------
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
246 1 AVAILABLE Recovery Appliance (RAHADR2)
RA_SBT_CDB122DR_3244939197_230_2kt4m80u_1_2_220
```

16.2.1.6 Adding Remaining Grants to the Upstream and Downstream Recovery Appliance

This task grants access to VPC users on both the upstream and downstream Recovery Appliances.

1. On RAHADR1, add the grant access to the one remaining VPC users.

```
SQL> exec dbms_ra.grant_db_access(
username => 'HADR_COMMON_VPCUSER',
db_unique_name => 'cdb122dr');

PL/SQL procedure successfully completed.
```

2. On RAHADR2, add the grant access to the two remaining VPC users. These users are pre-setup in the event that backups failover, due to RAHADR1 not being available.

```
SQL> exec dbms_ra.grant_db_access(
username => 'HADR_LOCAL_VPCUSER',
db_unique_name => 'cdb122dr');

PL/SQL procedure successfully completed.

SQL> exec dbms_ra.grant_db_access(
username => 'HADR_COMMON_VPCUSER',
db_unique_name => 'cdb122dr');

PL/SQL procedure successfully completed.
```

3. Verify the credentials are working correctly by logging into each target using only the credential alias. On each host run:

```
$ sqlplus /@dr_rahadr2

$ sqlplus /@dr_rahadr
```

16.2.1.7 Configuring Channel Device Parameters

This task configures the channel device parameters for use with the `DR_RAHADR` alias.

1. On one of the protected database hosts, run:

```
$ rman target / catalog /@dr_rahadr1

RMAN> CONFIGURE CHANNEL DEVICE TYPE 'SBT_TAPE' FORMAT '%d_%U' PARMS
"SBT_LIBRARY=/u01/app/oracle/product/12.2.0.1/dbhome_1/lib/libra.so,
ENV=(RA_WALLET='location=file:/u01/app/oracle/product/12.2.0.1/
dbhome_1/dbs/z
dlra credential_alias=dr_rahadr')";
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

2. (Optional) configure the following parameters, which are best practice recommendations.

```
RMAN> CONFIGURE BACKUP OPTIMIZATION on;

RMAN> CONFIGURE CONTROLFILE AUTOBACKUP on;

RMAN> CONFIGURE DEFAULT DEVICE TYPE TO sbt;

RMAN> CONFIGURE DEVICE TYPE SBT_TAPE PARALLELISM 2 BACKUP TYPE TO
BACKUPSET;

RMAN> CONFIGURE SNAPSHOT CONTROLFILE NAME TO '+RECOC1/cdb122dr/snapcf.f';

RMAN> CONFIGURE ARCHIVELOG DELETION POLICY TO backed up 1 times to device
type sbt;
```

16.2.2 Configuring Upstream and Downstream Recovery Appliance

This task creates host specific files for backups, loads the scripts on their respective hosts, and verifies the credentials.

1. On a host of the upstream Recovery Appliance, create the `backup_database_rahadr1.rman` text file with the following content.

```
{
allocate channel rahadr1_sbt_1 device type sbt
  format '%d_%U'
  PARMS="SBT_LIBRARY=/u01/app/oracle/product/12.2.0.1/dbhome_1/lib/
libra.so,
  ENV=(RA_WALLET='location=file:/u01/app/oracle/product/12.2.0.1/
dbhome_1/dbs/zdlra
  credential_alias=dr_rahadr1')";

allocate channel rahadr1_sbt_2 device type sbt
  format '%d_%U'
  PARMS="SBT_LIBRARY=/u01/app/oracle/product/12.2.0.1/dbhome_1/lib/
libra.so,
  ENV=(RA_WALLET='location=file:/u01/app/oracle/product/12.2.0.1/
dbhome_1/dbs/zdlra
  credential_alias=dr_rahadr1')";

backup
  tag '&1'
  cumulative incremental level 1
  filesperset 1
  section size 64g
  database
    plus archivelog
      not backed up
      filesperset 32
      delete input;
}
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

2. On a host of the downstream Recovery Appliance, create the `backup_database_rahadr2.rman` text file with the following content.

```
{
allocate channel rahadr2_sbt_1 device type sbt
  format '%d_%U'
  PARMS="SBT_LIBRARY=/u01/app/oracle/product/12.2.0.1/dbhome_1/lib/
libra.so,
  ENV=(RA_WALLET='location=file:/u01/app/oracle/product/12.2.0.1/
dbhome_1/dbs/zdlra
  credential_alias=dr_rahadr2')";

allocate channel rahadr1_sbt_2 device type sbt
  format '%d_%U'
  PARMS="SBT_LIBRARY=/u01/app/oracle/product/12.2.0.1/dbhome_1/lib/
libra.so,
  ENV=(RA_WALLET='location=file:/u01/app/oracle/product/12.2.0.1/
dbhome_1/dbs/zdlra
  credential_alias=dr_rahadr2')";

backup
  tag '&1'
  cumulative incremental level 1
  filesperset 1
  section size 64g
  database
    plus archivelog
      not backed up
      filesperset 32
      delete input;
}
```

3. Ensure the script on RAHADR1 does not exist by trying to delete it first. Then load the HADR1 script into the RAHADR1 Recovery Appliance.

```
$ rman target / catalog /@dr_rahadr1

RMAN> delete script backup_database;

RMAN> create script backup_database from file
'/home/oracle/backup_database_rahadr1.rman';
```

4. Ensure the script on RAHADR2 does not exist by trying to delete it first. Then load the HADR2 script into the RAHADR2 Recovery Appliance.

```
$ rman target / catalog /@dr_rahadr2

RMAN> delete script backup_database;

RMAN> create script backup_database from file
'/home/oracle/backup_database_rahadr2.rman';
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

5. Verify credentials have access to the database.

```
$ rman target / catalog /@dr_rahadr

RMAN> print script backup_database;

printing stored script: backup_database
{
allocate channel rahadr1_sbt_1 device type sbt
  format '%d_%U'
  PARMS="SBT_LIBRARY=/u01/app/oracle/product/12.2.0.1/dbhome_1/lib/
libra.so,
  ENV=(RA_WALLET='location=file:/u01/app/oracle/product/12.2.0.1/
dbhome_1/dbs/zdlra
  credential_alias=dr_rahadr1')";

allocate channel rahadr1_sbt_2 device type sbt
  format '%d_%U'
  PARMS="SBT_LIBRARY=/u01/app/oracle/product/12.2.0.1/dbhome_1/lib/
libra.so,
  ENV=(RA_WALLET='location=file:/u01/app/oracle/product/12.2.0.1/
dbhome_1/dbs/zdlra
  credential_alias=dr_rahadr1')";

backup
  tag '&1'
  cumulative incremental level 1
  filesperset 1
  section size 64g
  database
    plus archivelog
      not backed up
      filesperset 32
      delete input;
}
```

16.2.3 Backup Operation

This task starts the `backup_database` script.

The following RMAN command has to be executed in the protected database and should be used for all RMAN backup operations.

> **Note**
>
> When the script is run, the channel allocations indicate which Recovery Appliance is logged into and the Recovery Appliance database name.

1. Start RMAN

```
$ rman target / catalog /@dr_rahadr
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

2. Start the `backup_database` script. If RAHADR1 is running, the script logs into RAHADR1. Otherwise, the script logs into RAHADR2.

```
RMAN> run { execute script backup_database using 'Level1'; }

executing script: backup_database

allocated channel: rahadr1_sbt_1
channel rahadr1_sbt_1: SID=1936 instance=cdb122dr1 device type=SBT_TAPE
channel rahadr1_sbt_1: RA Library (RAHADR1)
SID=6DEA2A958DFBE0CFE05311F3850AB3AB

allocated channel: rahadr1_sbt_2
channel rahadr1_sbt_2: SID=394 instance=cdb122dr1 device type=SBT_TAPE
channel rahadr1_sbt_2: RA Library (RAHADR1)
SID=6DEA2A9CC2BBE0D0E05311F3850AC634
```

16.2.4 Real-Time Redo Transport

Real-Time Redo Transport for protected databases can be configured to regularly use the upstream Recovery Appliance, but to failover to the downstream Recovery Appliance when the upstream one isn't available. When the upstream Recovery Appliance becomes available again, redo transport automatically changes from using the downstream back to using the upstream.

16.2.4.1 Configuring the VPC User for Real-Time Redo Transport

This task establishes the VPC user for redo transport and then you choose between (1) enabling parameters in Data Guard Broker and (2) enabling log archive parameters.

1. Configure the `redo_transport_user` to the local VPC user.

```
$ sqlplus / as sysdba

SQL> alter system set redo_transport_user=hadr_local_vpcuser;

System altered.
```

2. Choose one of the two options.
   - Option 1: Use Data Guard Broker to Configure Real-Time Redo Transport
   - Option 2: Use log_archive* Parameters to Configure Real-Time Redo Transport

16.2.4.2 Option 1: Use Data Guard Broker to Configure Real-Time Redo Transport

This task enables Data Guard Broker parameters that establish failover of real-time redo transport from the upstream to the downstream Recovery Appliance.

1. Enable the `dg_broker*` parameters from a SQLPLUS session as `sysdba`.

```
$ sqlplus / as sysdba

SQL> alter system set
dg_broker_config_file1='+DATAC1/cdb122dr/dr1cdb122dr.dat';
System altered.
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
SQL> alter system set
dg_broker_config_file2='+DATAC1/cdb122dr/dr2cdb122dr.dat';
System altered.

SQL> alter system set dg_broker_start=true;
System altered.
```

2. Configure Data Guard Broker with respect to the primary databases, connection identifiers for the Recovery Appliances, network timeouts, and maximum number of failures. In the end, enable the configuration changes.

```
$ dgmgrl sys/myPassword

DGMGRL for Linux: Release 12.2.0.1.0 - Production on Tue Jun 5 11:37:44
2018

Copyright (c) 1982, 2017, Oracle and/or its affiliates. All rights
reserved.

Welcome to DGMGRL, type "help" for information.
Connected to "cdb122dr"
Connected as SYSDG.

DGMGRL> create configuration cdb122dr as primary database is cdb122dr
connect
identifier is '//scam06-scan3/cdb122dr';
Configuration "cdb122dr" created with primary database "cdb122dr"

DGMGRL> add recovery_appliance rahadr1 as connect identifier is
'dr_rahadr1';
Recovery Appliance "rahadr1" added

DGMGRL> add recovery_appliance rahadr2 as connect identifier is
'dr_rahadr2';
Recovery Appliance "rahadr2" added

DGMGRL> edit recovery_appliance rahadr1 set property MaxFailure=1;
Property "maxfailure" updated

DGMGRL> edit recovery_appliance rahadr1 set property ReopenSecs=10;
Property "reopensecs" updated

DGMGRL> edit recovery_appliance rahadr1 set property NetTimeout=8;
Property "nettimeout" updated

DGMGRL> edit recovery_appliance rahadr2 set property MaxFailure=1;
Property "maxfailure" updated

DGMGRL> edit recovery_appliance rahadr2 set property NetTimeout=8;
Property "nettimeout" updated

DGMGRL> edit database cdb122dr set property RedoRoutes = '(LOCAL : (rahadr1
async priority=1, rahadr2 async priority=2))';
Warning: ORA-16677: Standby database has the same or higher priority than
other members specified in the RedoRoutes group.
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
Property "redoroutes" updated

DGMGRL> enable configuration;
Enabled.
```

> **Note**
>
> If Redo Transport does not start, then you may need to bounce the protected database. For a RAC database, this can be done in a rolling fashion.

16.2.4.3 Option 2: Use log_archive* Parameters to Configure Real-Time Redo Transport

This task enables manually changes several `log_archive*` parameters that establish failover of real-time redo transport from the upstream to the downstream Recovery Appliance.

- Log into `sqlplus` as `rasys/ra` and change several parameters with respect to the primary databases, connection identifiers for the Recovery Appliances, network timeouts, and maximum number of failures. In the end, enable the configuration changes.

```
$ sqlplus rasys/ra

SQL> alter system set log_archive_config =
'dg_config=(cdb122dr,rahadr1,rahadr2)';

SQL> alter system set log_archive_dest_2='service=dr_rahadr1 ASYNC NOAFFIRM
delay=0 optional compression=disable max_failure=1 max_connections=1
reopen=10 db_unique_name=rahadr1 net_timeout=8 group=1 priority=1
valid_for=(online_logfile,all_roles)';

SQL> alter system set log_archive_dest_3='service=dr_rahadr2 ASYNC NOAFFIRM
delay=0 optional compression=disable max_failure=1 max_connections=1
reopen=300 db_unique_name=rahadr2 net_timeout=8 group=1 priority=2
valid_for=(online_logfile,all_roles)';

SQL> alter system set log_archive_dest_state_2=enable;

SQL> alter system set log_archive_dest_state_3=enable;
```

> **Note**
>
> If Redo Transport does not start, then you may need to bounce the protected database. For a RAC database, this can be done in a rolling fashion.

16.3 Replication Mode for HADR

Demonstrates a high availability disaster recover scenario involving replicated recovery appliances.

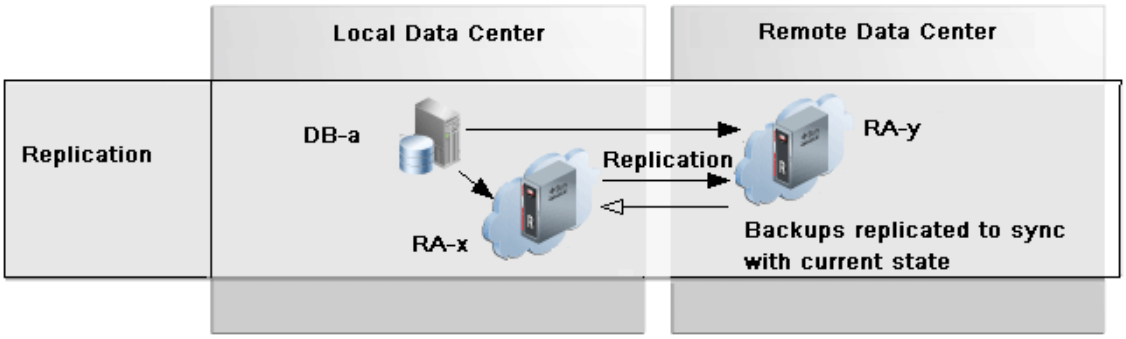
Figure 16-1 shows two data centers, one local and one remote, each with a recovery appliance RA-x and RA-y that are configured as a replication pair, or bi-directional replication.

Copyright © 2014, 2026, Oracle and/or its affiliates.

The database in the local data center sends its backups and redo logs to RA-x, as usual. RA-x then replicates the backups and redo logs to RA-y in the remote local data center. If local RA-x goes offline, the backups and redo are redirected to remote RA-y, with full recoverability to both data centers. When RA-x comes back online, remote RA-y replicates backups to local RA-x to get it in sync with the current state.

In this example, backups from the remote data center's databases to RA-y are not replicated to RA-x in the local data center.

Figure 16-1 Backups Replicated to two Recovery Appliances



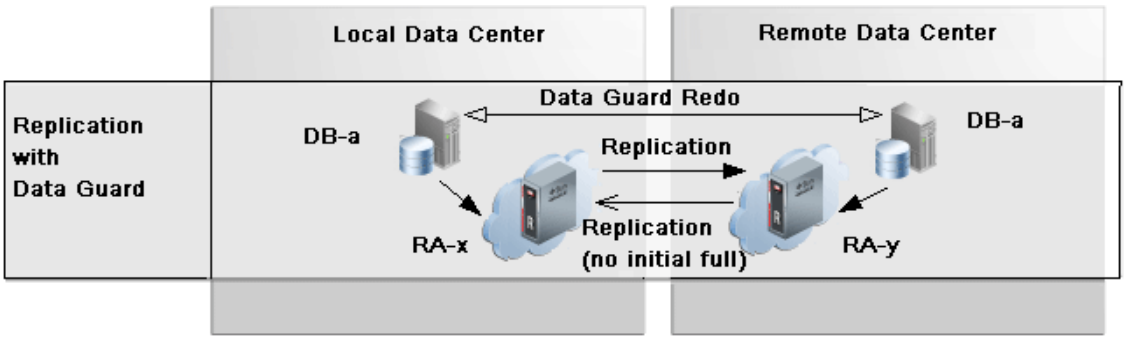
16.4 Backup Anywhere Mode for Data Guard

Demonstrates a how Backup Anywhere supports Data Guard.

Figure 16-2 shows two data centers for local and remote, where the primary site (upstream) is local RA-x and the standby site (downstream) is remote RA-y. The database in the local data center sends its primary backups and logs to RA-x, as usual.

When a failover or switchover is carried out, the remote database becomes the new primary and will backup and send redo to RA-y. Meanwhile a reversal of the replication automatically occurs from RA-y to RA-x for the backups and logs. All backups needed to synchronize RA-y are replicated from RA-x, while new primary backups and redo to RA-y replicate as normal to RA-x. No initial full backup is replicated, which reduces WAN consumption.

Figure 16-2 Backup Anywhere Mode for Data Guard



A common use case with Data Guard and taking backups only on one database, is when the primary database has production workloads running that backups might impact. Instead,

Copyright © 2014, 2026, Oracle and/or its affiliates.

backups are taken from the standby database to its recovery appliance (RA-y). Replication from RA-y to RA-x keeps it in synch.

16.5 Request_Only Mode for Data Guard

Demonstrates a how replication `request_only` mode supports Data Guard.

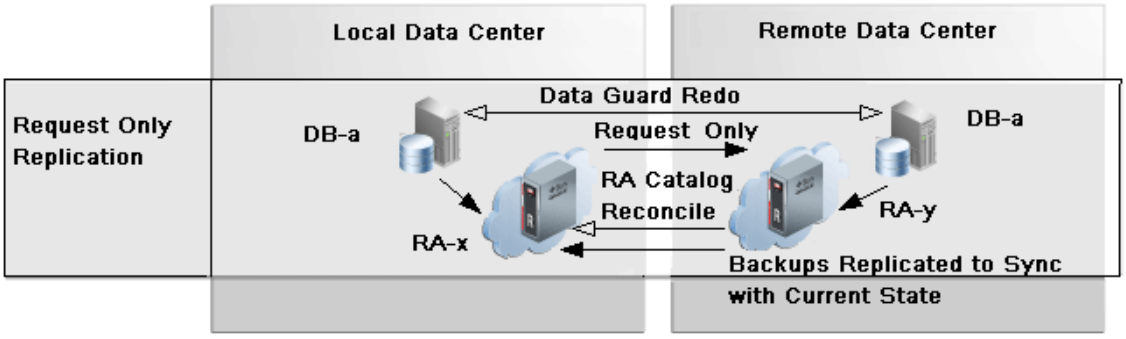
The purpose of the `request_only` mode is to allow a recovery appliance to request backups from the second Recovery Appliance of a pair, in order to fill gaps in its backups, such as after an offline period. There is no active replication between the upstream and downstream. Replication only happens when backups are requested by the upstream RA-x to fill gaps after an outage. Data Guard and redo logs keeps the remote database in sync with the local database.

Figure 16-3 shows two data centers for local and remote, where the primary site is local RA-x and the standby site is remote RA-y. Backups are taken at primary and standby databases, each to their respective local Recovery Appliances. In `request_only` mode, replication traffic does not occur from RA-x to RA-y. Dataguard and redo logs keeps the remote database in sync with the local database.

The replication servers are configured for bi-direction replication. When using `add_replication_server`, RA-y gets a protection policy as normal. However, `add_replication_server` for RA-x has the protection policy with `REQUEST_ONLY=TRUE`.

When a switchover from the primary (local) to the secondary (remote) has happened, replication traffic does not occur from RA-x to RA-y. However, the RA-x catalog is kept in sync with the RA-y backups. When RA-x is offline, standby backups continue to be sent to RA-y with full recoverability to both data centers. When RA-x comes back online, RA-x requests missing backups from RA-y in order to sync with the current state.

Figure 16-3 Replication Request Mode of Backup Anywhere



16.6 Replication Read-Only Mode when Migrating to New Data Center

Demonstrates a how replication read-only mode supports migrating from one data center to another, and how existing backups on a downstream recovery appliance can be accessed read-only by an upstream recovery appliance.

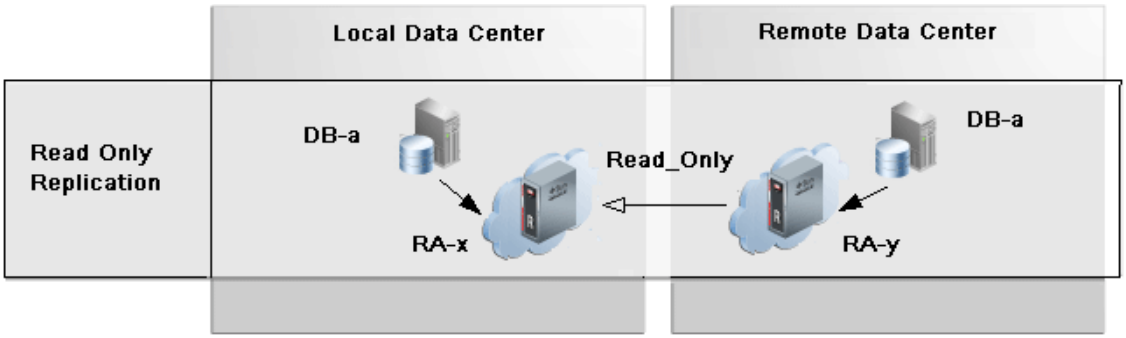
Figure 16-4 shows the databases in local data center backing up to its RA-x. In the example, all of the local databases need to be moved to the remote data center, and RA-x is to be decommissioned.

Copyright © 2014, 2026, Oracle and/or its affiliates.

The databases in the remote data center are created by cloning them from RA-x. Then a replication server is created on RA-y as the upstream, and RA-x as the downstream. On RA-y, a protection policy is added to the replication server with `READ_ONLY=TRUE`.

The databases in the remote data center start backing up to RA-y. Should a database need to be recovered, backups on local data center RA-x remain accessible through RA-y until RA-x's backup's are obsolete and RA-x is decommissioned.

Figure 16-4 Read-Only Mode of Backup Anywhere



Copyright © 2014, 2026, Oracle and/or its affiliates.

17

Monitoring the Recovery Appliance

This chapter explains how to perform basic monitoring of a Recovery Appliance, including configuring the metric and configuration settings.

17.1 About Monitoring the Recovery Appliance

This section contains the following topics:

- Purpose of Monitoring the Recovery Appliance
- Overview of Recovery Appliance Monitoring Capabilities
- Cloud Control Interface for Monitoring the Recovery Appliance
- Basic Tasks for Monitoring the Recovery Appliance

> **See Also**
>
> "Protection Policies" for an architectural overview

17.1.1 Purpose of Monitoring the Recovery Appliance

A crucial part of ongoing Recovery Appliance administration is regularly monitoring the overall health of the Recovery Appliance, and checking the status of protected databases, backup and replication jobs, and storage usage.

17.1.2 Overview of Recovery Appliance Monitoring Capabilities

This section describes the monitoring tools supplied by Oracle.

17.1.2.1 Cloud Control

The primary monitoring tool for Recovery Appliance administrators is the Oracle Enterprise Manager Cloud Control (Cloud Control) incident and event notification framework. The primary interface is the Recovery Appliance home page, which prominently displays warnings, alerts, and errors. The monitoring framework integrated with Cloud Control is an effective way of managing issues and tracking them until resolution.

Space management is a crucial part of administering the Recovery Appliance. To have sufficient time to accommodate storage demands, you must know when estimated storage needs are approaching the amount of total storage available. Cloud Control provides warnings and error messages regarding aggregate storage usage, providing ample time to make necessary changes.

Cloud Control enables you to customize settings to meet your management goals. For example, you can receive warnings if the space needed to meet the recovery window goal of a specific database is a user-specified percentage of its reserved space. You can also configure

Copyright © 2014, 2026, Oracle and/or its affiliates.

email alerts so that you receive immediate notification of issues without having to log in to the system.

17.1.2.2 Oracle Configuration Manager

Oracle Configuration Manager collects configuration information (by default, every day) and uploads it to the Oracle Management Repository. If you log a service request, then the configuration data is associated with the service request. Oracle Support Services can analyze the data and provide better service.

Benefits of Oracle Configuration Manager include the following:

- Reduces time for resolution of support issues
- Provides pro-active problem avoidance
- Improves access to best practices and the Oracle knowledge base
- Improves understanding of customer's business needs and provides consistent responses and services

Oracle Configuration Manager software is installed in each Oracle home. Typically, each Oracle home has a collector configured that gathers and uploads information under its My Oracle Support (MOS) credentials. You can also configure a central collector, which gathers information for the Oracle home in which it resides *and* Oracle homes in which the collector is disconnected or not configured.

17.1.2.3 Auto Service Request (ASR)

Auto Service Request (ASR) is a feature that automatically opens service requests when specific Recovery Appliance hardware faults occur. ASR detects faults in the most common server components, such as disks, fans, and power supplies. ASR monitors only server components and does not detect all possible faults.

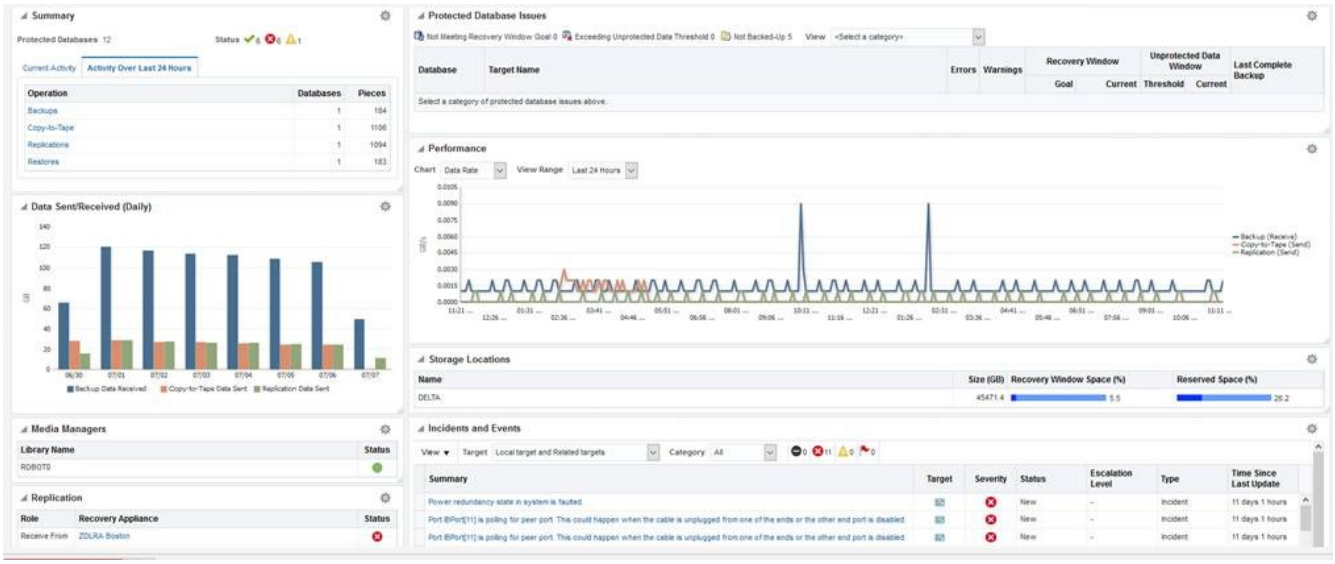
ASR is not a replacement for other monitoring mechanisms, such as SMTP and SNMP alerts, within the customer data center. It is a complementary mechanism that expedites and simplifies the delivery of replacement hardware.

> **See Also**
>
> *Zero Data Loss Recovery Appliance Owner's Guide* to learn how to set up ASR

17.1.3 Cloud Control Interface for Monitoring the Recovery Appliance

The primary interface for monitoring the Recovery Appliance is the Recovery Appliance Home page. The Home page lists any existing warnings and alerts, as shown in the following graphic:

Copyright © 2014, 2026, Oracle and/or its affiliates.



The following sections of the Home page show monitoring information:

- Summary

  This section shows the number of databases with no issues, with alerts, and with warnings. In Cloud Control, an alert is an indicator that a particular metric condition has been encountered. For example, an alert might indicate that a metric threshold has been reached.

- Media Managers and Replication

  These sections show the status of copy-to-tape and Recovery Appliance replication services.

- Protected Database Issues

  This section summarizes the backup status for protected databases, and provides a category filter so you can view which databases are affected.

- Incidents and Events

  This section displays incidents and events reported for the Recovery Appliance and all associated targets. You can filter by target and category. You can click the Summary link to drill down to the Incident Manager to view detailed information about the incident.

> **Note**
>
> Warnings automatically clear when the underlying issue is resolved.

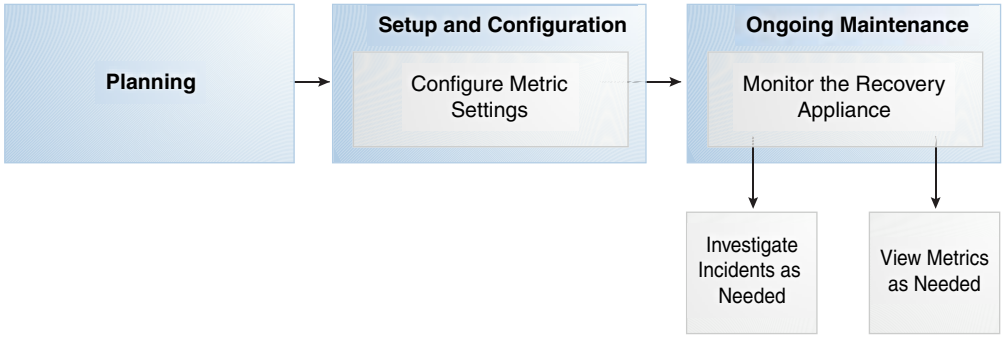
> **See Also**
>
> - "Accessing the Recovery Appliance Home Page"

Copyright © 2014, 2026, Oracle and/or its affiliates.

17.1.4 Basic Tasks for Monitoring the Recovery Appliance

This section explains the basic tasks involved in monitoring the Recovery Appliance. The following diagram shows the overall workflow described in Recovery Appliance Workflow , with the monitoring tasks highlighted.

Figure 17-1 Monitoring Tasks in the Recovery Appliance Workflow



Typically, you perform monitoring tasks in the following sequence:

1. During the configuration phase (see "Setup and Configuration for Recovery Appliance"), configure your metric settings. For example, you may want to configure the Recovery Appliance to issue a warning if a threshold is passed.

   "Modifying the Metric and Collection Settings" describes this task.

2. During the ongoing maintenance phase (see "Maintenance Tasks for Recovery Appliance"), modify protection policies as needed. Typical modification tasks include:
   - Investigate incidents as needed.

     "Viewing the Incident Manager Page" describes this task.
   - View metrics as needed.

     "Modifying the Metric and Collection Settings" describes this task.

17.2 Modifying the Metric and Collection Settings

The Metric and Collection Settings page provides details about thresholds and schedules for target metric collection. Using this page, you can edit the warning threshold and critical threshold values of target metrics and other collected items, and the time intervals for collection. The page shows a pencil icon in the Edit column for modifiable settings.

Prerequisites

You must log in to the Recovery Appliance metadata database as RASYS.

Assumptions

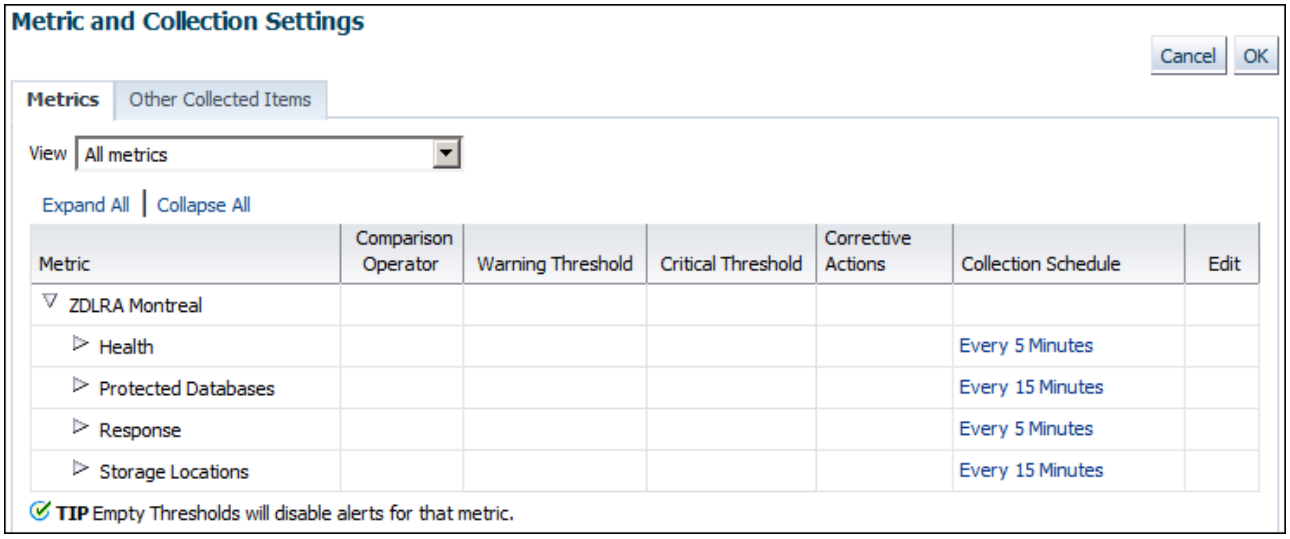
You want to receive warnings when the space needed to meet the recovery window goal for a database is 80% percent of its reserved space setting. You want the critical threshold to be 95%.

Copyright © 2014, 2026, Oracle and/or its affiliates.

To modify the metric and collection settings:

1. Access the Recovery Appliance Home page, as described in "Accessing the Recovery Appliance Home Page".
2. From the **Recovery Appliance** menu, click **Monitoring**, and then click **Metric and Collection** settings.

   The Metric and Collection Settings page appears.



3. If **All metrics** is not selected in the **View** menu, then select it.

   The page refreshes to show all the available metrics.
4. Expand **Protected Databases**.
5. Scroll down the page until you find the row that says `Recovery Window Space as a Percentage of Reserved Space`.
6. For this row, enter the following values, and then click **OK**:
   - In the Warning Threshold column, enter `80`.
   - In the Critical Threshold column, enter `95`.

   A confirmation message appears.

> **Note**
>
> To change the default text of the alert message that is generated when these thresholds are passed, click the pencil icon.

7. Modify other metric settings as needed.

17.3 Viewing the Incident Manager Page

The Incidents and Events section shows all incidents, events, and warnings for a Recovery Appliance. Click any incident to open the Incident Manager page. Incident Manager provides, in one location, the ability to search, view, manage, and resolve incidents and problems impacting your environment.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Prerequisites

You must log in to the metadata database as RASYS.

Assumptions

This tutorial assumes that Incidents and Events section of the Recovery Appliance Home page for your Recovery Appliance shows a warning. You want to get more details about it.

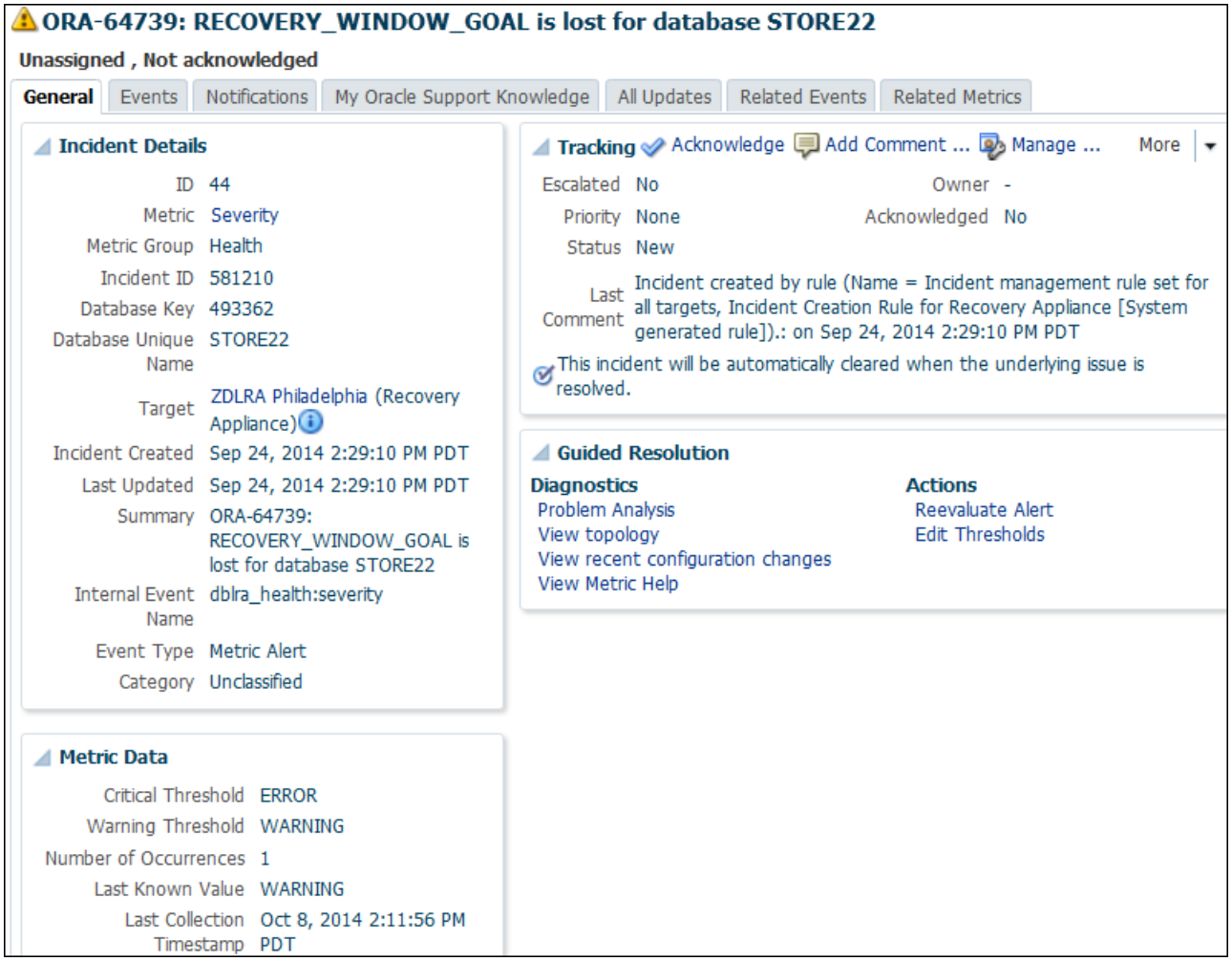
To view the Incident Manager page:

1. Access the Recovery Appliance Home page, as described in "Accessing the Recovery Appliance Home Page".
2. Review the Incidents and Events section for possible problems.

   For example, the section shows the following warning:

   ```
   ORA-64739: RECOVERY_WINDOW_GOAL is lost for database STORE22
   ```
3. Click the summary link of the incident that you are interested in.

   The Incident Manager page for the selected warning appears, with the General subpage selected:



Copyright © 2014, 2026, Oracle and/or its affiliates.

4. Click the subpages to get detailed information about the incident.

17.4 Monitoring Performance

Recovery Appliance ships with two utilities—rastat.pl and network_throughput_test.sh—that can assist you in evaluating the performance of your system.

17.4.1 Generating Performance Statistics by Using the rastat Utility

`rastat.pl` is a command line utility that runs tests against the Recovery Appliance to gather performance statistics which can help you identify system bottlenecks.

The tests can generate statistics on:

- backup data sent to the Recovery Appliance over the network
- restore data received from the Recovery Appliance over the network
- Recovery Appliance ASM disk group read or write I/O
- Recovery Appliance container file read or write I/O
- Recovery Appliance container file allocation rate

The utility is a Perl script that can be run from any Linux or Unix-based client machine that is either a protected database or an upstream Recovery Appliance. The I/O tests however, can also be run directly from the Recovery Appliance server.

You can run multiple tests in parallel on one or more protected databases to simulate a real environment. Each test result represents the performance of an individual client. Note that ongoing activities between other protected databases and the Recovery Appliance being tested, such as backup and restore or other testing, can impact the resulting statistics.

17.4.1.1 Prerequisites for Running the rastat Utility

Before you run the rastat utility, ensure that the following requirements are met:

- The platform on which you will be running rastat is either Linux or Unix.
- If you will be running the utility from a protected database, copy the `rastat.pl` file from the `/opt/oracle.RecoveryAppliance/client/` directory of a Recovery Appliance compute server to the protected database.
- Complete the steps to enroll the protected database with the Recovery Appliance as described in "Enrolling Protected Databases".
- Ensure that the `$ORACLE_HOME` and `$ORACLE_SID` environment variables are configured if you do not plan to set them by using the applicable options when you run the utility.

17.4.1.2 Running the rastat Utility

This section describes how to run rastat.pl and provides several examples of how to execute various performance tests, along with sample output.

Copyright © 2014, 2026, Oracle and/or its affiliates.

> **Note**
>
> If the NETBACKUP and NETRESTORE tests do not display the results to the standard output, you can view results by looking at the sbtio<pid>.log files.

To run the rastat utility:

1. Ensure that the system from which you are running the utility meets the requirements, as described in "Prerequisites for Running the rastat Utility".
2. Open a command prompt window.
3. Enter the applicable command syntax for the tests you want to run, and press Enter.

   Refer to the "rastat Utility Reference " for information about the general syntax and the options for each test.

Example 1: Running rastat to Test Backup Performance

In the following example, the NETBACKUP test is specified, the backup file size is set to 2048 megabytes, the Recovery Appliance VPC user connection string is supplied, and the RMAN configuration is set by using the --parms option.

```
>$ORACLE_HOME/perl/bin/perl rastat.pl --test=NETBACKUP --filesize=2048M
--catalog=rman/rman@inst2 --parms='SBT_LIBRARY=/u01/oracle/lib/libra.so,
ENV=(RA_WALLET=location=file:/u01/oracle/dbs/ra_wallet
credential_alias=ra-scan:1521/zdlra5:dedicated)'

NETWORK TEST FROM PROTECTED DATABASE TO RECOVERY APPLIANCE

393 MB/s, 2048 MB sent
```

Example 2: Running rastat to Test I/O Reads from a Recovery Appliance ASM Disk Group

In the following example, the ASMREAD test is specified, the test file size is set to 2048 megabytes, the Recovery Appliance SYS user connection string is supplied, and +RCVAREA is specified as the disk group to read from.

```
>$ORACLE_HOME/perl/bin/perl rastat.pl --test=ASMREAD --filesize=2048M
--rasys=admin/admin@inst2 --diskgroup=+RCVAREA

RECOVERY APPLIANCE READ IO TEST FROM DISK

Disk Group: +RCVAREA

2048 MB, 6.06s read IO time, .65s CPU time, 337.99 MB/s, 10.79% CPU usage

PL/SQL procedure successfully completed.
```

Example 3: Running rastat to Test I/O Writes to a Recovery Appliance Container Group

In the following example, the CONTAINERWRITE test is specified, the test file size is set to 2048 megabytes, the Recovery Appliance SYS user connection string is supplied, and the BLOCK_POOL container group is specified as the disk group to write to.

```
>$ORACLE_HOME/perl/bin/perl rastat.pl --test=CONTAINERWRITE --filesize=2048M
--rasys=admin/admin@inst2 --diskgroup=/:BLOCK_POOL

RECOVERY APPLIANCE WRITE IO TEST TO CONTAINER FILES
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
Disk Group: /:BLOCK_POOL

2048 MB, 9.55s write IO time, 3.50s CPU time, 214.35 MB/s, 36.60% CPU usage

PL/SQL procedure successfully completed.
```

Example 4: Running rastat to Test File Allocation to a Recovery Appliance Container Group

In the following example, the `CONTAINERALLOC` test is specified, the test file size is set to 2048 megabytes, the Recovery Appliance SYS user connection string is supplied, and the `BLOCK_POOL` container group is specified as the disk group to write to.

```
>$ORACLE_HOME/perl/bin/perl rastat.pl --test=CONTAINERALLOC --filesize=2048M
--rasys=admin/admin@inst2 --diskgroup=/:BLOCK_POOL

RECOVERY APPLIANCE CONTAINER FILE ALLOCATION TEST

Disk Group: /:BLOCK_POOL

2048 MB, 6.24s allocation time, 3.69s CPU time, 328.34 MB allocated per second, 59.09%
CPU usage

PL/SQL procedure successfully completed.
```

17.4.2 Testing Network Throughput

You can measure theoretical network throughput in a Recovery Appliance environment by using the network_throughput_test.sh script that ships with the appliance.

See My Oracle Support Note Doc ID KB373398 for information and instructions on how to use the utility.

Copyright © 2014, 2026, Oracle and/or its affiliates.

18

Accessing Recovery Appliance Reports

The pre-created Oracle Business Intelligence (BI) Publisher reports specific to the Recovery Appliance have been migrated from Oracle Enterprise Manager Cloud Control to Oracle Analytics Publisher and consolidated with other analytic reports.

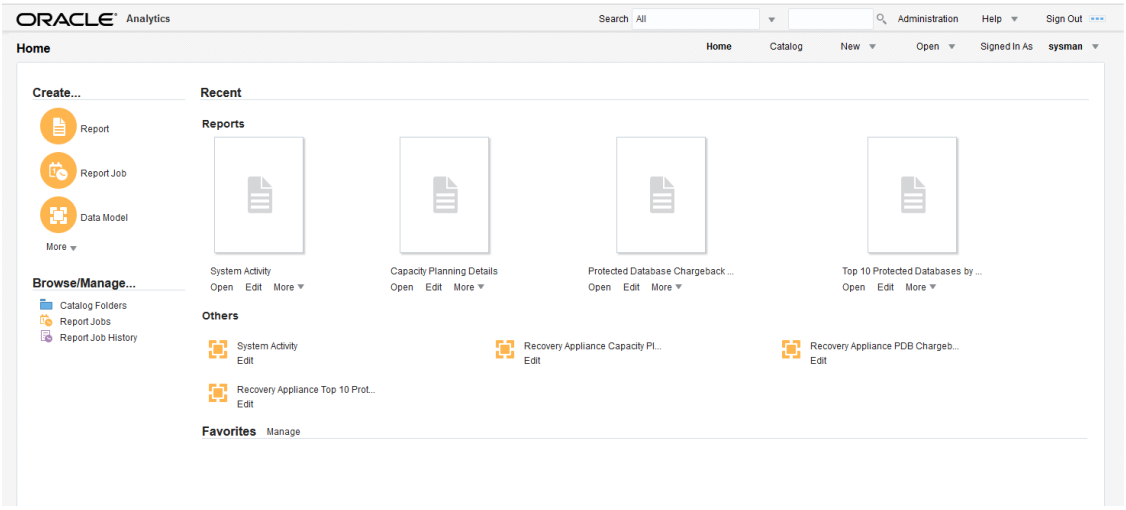
Oracle Analytics Publisher is available as component of the **Complete Suite of Oracle Analytics Server (OAS),** or as an independently-installed **Oracle Analytics Publisher Component**. When the OAS suite is used, the catalog used by OAS is also used as the Oracle Analytics Publisher repository.

Oracle Analytics Publisher is accessed using an environment-specific version of the following generic URL.

```
http(s)://<host>:<port>/xmlpserver
```

With Oracle Analytics, you can now perform collaborative analytics for users and their enterprises, built on a high-performance platform with flexible data storage.

Figure 18-1 Oracle Analytics Publisher Home Screen



18.1 Purpose of Recovery Appliance Reports

A principal task for a Recovery Appliance administrator is storage capacity planning. Through *Oracle Analytics Publisher* (OAP), Recovery Appliance provides pre-created OAP reports that enable you to meet the following goals:

- Ensure that the Recovery Appliance has sufficient storage space for its needs

  By using the capacity reports, you can plan for additional storage, reduce the number of new protected databases added to the Recovery Appliance, or adjust protection policies so that the aggregate recovery window space decreases.

Copyright © 2014, 2026, Oracle and/or its affiliates.

- Ensure that the network is not overloaded

  The capacity reports also indicate whether the Recovery Appliance has maximized network capacity. In some cases, you can reduce network load by redistributing network traffic more evenly throughout the day. If network traffic is not distributed, and if network peaks are close to maximizing network bandwidth, then you may need to adjust the backup window times of some protected databases.

- Provide a good view of system performance and activity for service requests
- Obtain a brief or highly detailed status report for any protected database, which can sometimes be useful for troubleshooting databases that are not meeting recovery window goals

18.2 Pre-Created Oracle Analytics Publisher Reports

Recovery Appliance provides the following pre-created *Oracle Analytics Publisher* (OAP) reports. [Enterprise Manager 13.4 and earlier, these reports were available within Cloud Control as *BI Publisher* reports.]

- **`Capacity Planning Details`:** This report provides a finer granularity of information about capacity planning. It provides information on storage capacity, network throughput, CPU utilization for each protected database host, and disk and flash storage I/O throughput over time. Unlike the capacity planning summary, the detailed report also has memory and IOPS summary information, and detailed daily data.
- **`Capacity Planning Summary`:** This report provides an overview of storage utilization and aggregated network traffic for the Recovery Appliance so that you can forecast when it will run out of capacity. Of special usefulness is the summary table, which provides a quick view of the number of days until capacity is exceeded. The network capacity planning summary provides a view of the aggregated network traffic over various time periods. This view shows both average and maximum rates, which are based on network samples.
- **`Protected Database Chargeback Greatest`:** This report is used for chargeback of protected databases enrolled with the Reovery Appliance. With this model, the consumer pays for the entire recovery window space needed upfront.
- **`Protected Database Chargeback Least`:** This report is used for chargeback of protected databases enrolled with the Recovery Appliance. With this model, the consumer pays only for the utilized space.
- **`Protected Database Details`:** This report provides extensive information about a protected database including summaries of the following:
  - The protection policy
  - Recovery Appliance storage location
  - The disk recovery window goal
  - The reserved space, which is the minimum amount of disk space in the Recovery Appliance reserved for the database to meet its disk recovery window goal
  - The status of real-time redo transport, which eliminates data loss
  - Data sent and received over time for backup, copy-to-tape, and Recovery Appliance replication operations, which gives a good overview of the traffic coming to and from the protected database to the Recovery Appliance
- **`Recovery Window Summary`:** This report provides a list of protected databases that are not meeting their recovery window goal or are exceeding their exceeding their unprotected window threshold (see `unprotected_window` in "`CREATE_PROTECTION_POLICY`"). You can use this report as a quick view of recovery window and unprotected data window issues for

Copyright © 2014, 2026, Oracle and/or its affiliates.

a Recovery Appliance, and then follow up on individual protected databases using the Protected Database Details report.

- **`System Activity`:** This report is a diagnostic aid for problems with the Recovery Appliance system and provides an organized visual report of all activity on the system. This includes general state, storage usage, running tasks, incidents encountered, and API commands executed.

  There are seven (7) metrics which the system activity uses. These metrics are disabled out of the box.

  The metrics can be enable inside of Enterprise Manager from the **ZLDRA** homepage.

  1. Select the menu **Recovery Appliance->Monitoring->Metric and Collection Settings**
  2. Click on the **Other Collected Items** tab.
  3. Click on the **Disabled** link in the **Collection Schedule** for the **Systems Activity: Active Session** row.
  4. Click the **Enable** button and then continue.

     Note that **Systems Activity: Active Session**, **Systems Activity: Protected Database Datafile Details and Systems Activity**: **Replication Servers** are each enabled.
  5. Click on the **Disabled** link in the **Collection Schedule** for the **Systems Activity: API History** row.
  6. Click the **Enable** button and then continue.

     Note that the rest of the **Systems Activity** metrics are now enabled.

- **`Top 10 Protected Databases by Data Transfer`:** This report ranks the top 10 protected databases according to the amount of backup data transferred to or from the Recovery Appliance. The report aggregates data by hour or by day. Specifically, the report measures the following amounts:
  - Backup data sent to the Recovery Appliance
  - Replication data sent by the Recovery Appliance
  - Copy-to-tape data sent by the Recovery Appliance

  This report does not correlate directly to how much space is being used by backups of the ranked databases.

> **Note**
>
> Although performance data is accessible through SQL queries of `V$` and recovery catalog views, Oracle highly recommends that you use the OAP reports instead. Recovery Appliance has finite CPU and other resources, so if users run frequent or expensive SQL queries, then overall Recovery Appliance performance can suffer.

> **See Also**
>
> - Summary of Recovery Appliance Views for a complete list of all the Recovery Appliance views
> - Oracle Enterprise Manager Licensing Information

Copyright © 2014, 2026, Oracle and/or its affiliates.

18.3 Accessing the Recovery Appliance Reports Page in Cloud Control

This section explains how to access the Recovery Appliance Reports page, which links to all pre-created reports.

- EM_13.5
- EM_13.4

EM_13.5

To access the OAP reports for the Recovery Appliance:

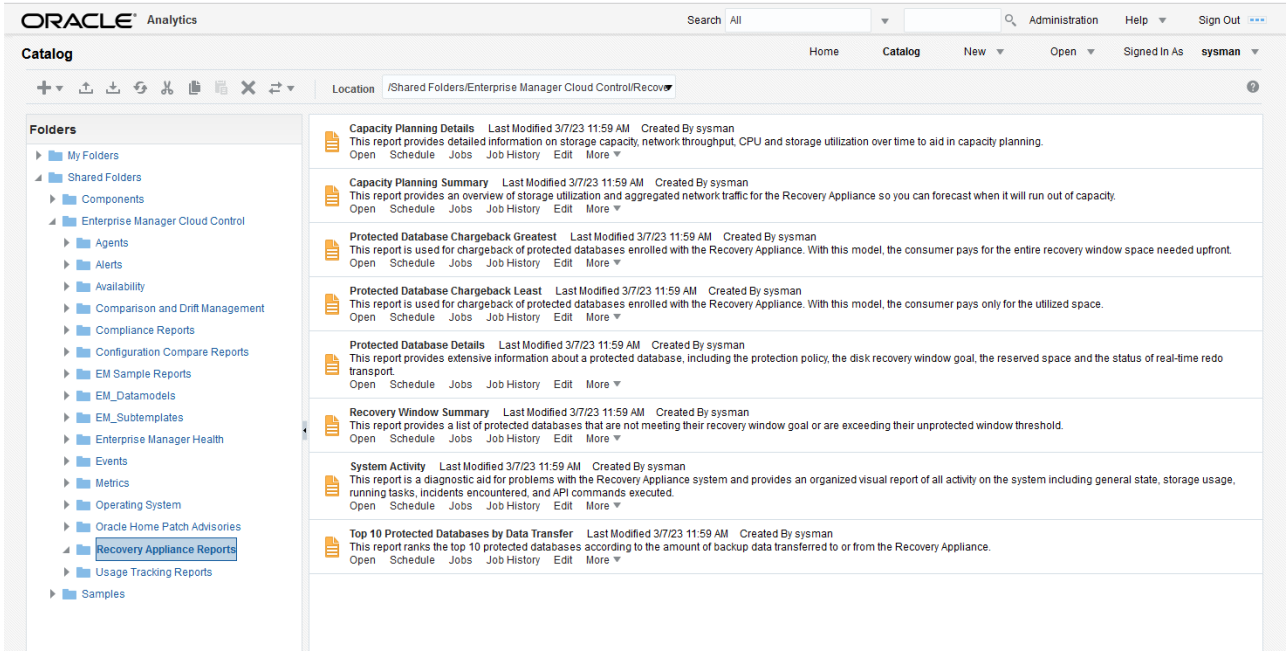
1. Start Oracle Analytics Publisher in your environment using specific version of the following generic URL.

   ```
   http(s)://<host>:<port>/xmlpserver
   ```

2. Select `Catalog` from the navigation items in the upper right of OAP.
3. In the `Folders` tree, navigate into the tree.

   ```
   Shared Folders -> Enterprise Manager Cloud Control -> Recovery Appliance Reports
   ```

Figure 18-2 Oracle Analytics Catalog



Copyright © 2014, 2026, Oracle and/or its affiliates.

EM_13.4

EM 13.4: To access the Recovery Appliance Reports page:

1. Access the Recovery Appliance Home page, as described in "Accessing the Recovery Appliance Home Page".
2. From the **Enterprise** menu, select **Reports**, and then **BI Publisher Enterprise Reports**.

   The BI Publisher Enterprise Reports page appears.
3. Expand the **Enterprise Manager Cloud Control** folder, and then expand the **Recovery Appliance Reports** subfolder.

   Links to the pre-created reports are shown.

18.4 Oracle Analytics Publisher Report Scheduling

Recovery Appliance provides the following pre-created *Oracle Analytics Publisher* (OAP) reports, formerly *BI Publisher*:

Oracle recommends that you configure OAP to generate reports automatically on a regular schedule (for example, every week), and to send the reports by email to the backup management team. You can also generate reports as needed using the techniques described in this chapter.

To schedule a report for the Recovery Appliance in OLP:

1. In OLP catalog, navigate to where the Recovery Appliance reports are: `Shared Folders -> Enterprise Manager Cloud Control -> Recovery Appliance Reports`.
2. Of the reports listed, each has actions listed under their descriptions: `Open`, `Schedule`, `Jobs`, `Job History`, and more.
3. For a given report underneath its description, select the `Schedule` action.

   .
4. The `Schedule Report Job` page has several screens of information.
   - **`General` tab:** provides the path to the report and a drop down to select the Recovery Appliance that the report is run against.
   - **`Output` tab:** provides checkboxes for `Make Output Public`, `Save Data for Republishing`, and `Compress output prior to delivery`. The `Output` area defines the name, format (HTML, PDF, Excel .xlsx), locale, time zone, etc. for a given output product. The `Destinations` area defines where the reports are sent.
   - **`Schedule` tab:** provides controls for determining the job frequency, if it should be run now or at some future time. The `Use Trigger` allows you to conditionally run an occurrence of the job. When the scheduled time occurs, the trigger is checked. If the trigger returns data, the job proceeds; if not data is returned, the job is skipped. Additional controls are can be established for the trigger, such as retry limit, pause time, etc.
   - **`Notification` tab:** shows the email or HTTP servers to receive the generated report.
   - **`Diagnostic` tab:** displays checkbox items several diagnostic tools: `Enable SQL Explain Plan`, `Enable Data Engine Diagnostic`, `Enable Report Processor Diagnostic`, `Enable Consolidated Job Diagnostic`

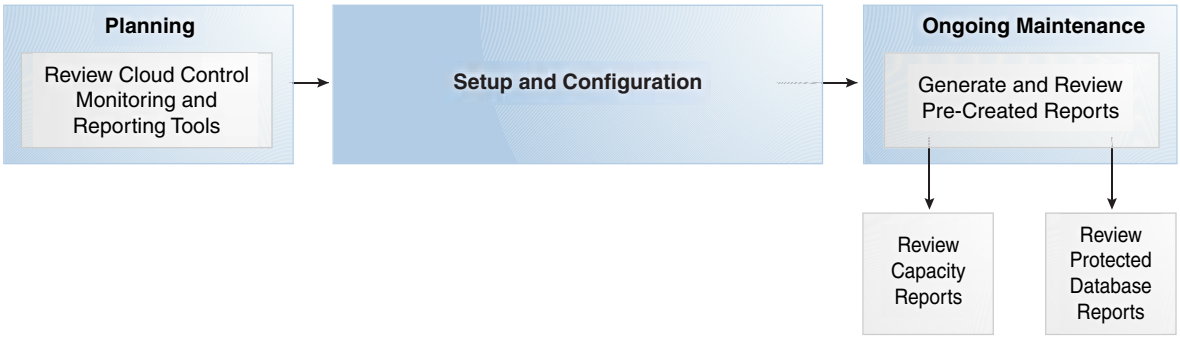
Copyright © 2014, 2026, Oracle and/or its affiliates.

> **See Also**
>
> Configure the Scheduler to learn how to schedule reports with Oracle Analytics Publisher in Oracle Analytics Server.

18.5 Basic Tasks for Accessing Recovery Appliance Reports

This section explains the basic tasks involved in managing reports. Figure 18-3 shows the overall workflow described in Recovery Appliance Workflow , with the reporting tasks highlighted.

Figure 18-3 Reporting Tasks in the Recovery Appliance Workflow



Typically, you perform reporting tasks in the following sequence:

1. During the planning phase, familiarize yourself with the monitoring and reporting tools available through Cloud Control.

   "Planning for Recovery Appliance" describes these tasks.

2. During the ongoing maintenance phase (see "Maintenance Tasks for Recovery Appliance"), review the reports as needed. Typical tasks include:

   - Review the Storage Capacity Planning Summary every week, using the Capacity Planning Details to get more detailed information.
   - Review the protected database reports as needed:

Copyright © 2014, 2026, Oracle and/or its affiliates.

Part II
Recovery Appliance Life-Cycle

An important part of the Recovery Appliance life-cycle is keeping its software up to date with relevant patches and upgrades.

Copyright © 2014, 2026, Oracle and/or its affiliates.

19

Best Practices for the Recovery Appliance

This chapter provides several practices that assist in optimally configuring the Recovery Appliance.

Oracle recommends following the operational best practices below when using Oracle Zero Data Loss Recovery Appliance, because they can help prevent issues that might impact performance, availability, or security.

- Modifying the Recovery Appliance configuration is limited. For details on specific restrictions and allowed configuration exceptions, refer to MOS note 2172842.1.
- Best practice for database protection is to enable real-time redo to Recovery Appliance, in which redo is automatically converted to archived log backups.

  If RMAN archive log backups are used instead, LOW compression algorithm is recommended for optimal balance of compression time and space savings.
- To ensure a stable and certified software environment, follow the recommended Recovery Appliance (RA) software configurations outlined in published My Oracle Support (MoS) note- **KB202620, PNEWS2275.**.

  Before applying patches, review the 'Zero Data Loss Recovery Appliance Supported Versions' document (Doc ID 1927416.1) to avoid known critical issues and verify compatible software stack combinations.
- When removing databases from the Recovery Appliance, delete them one at a time, starting with the smallest database backups and progressing to the larger ones. Avoid submitting multiple delete requests simultaneously.
- Recovery Appliance operates with an initial Level 0 backup and Level 1 incremental backups thereafter. Refrain from taking multiple L0 backups, unless instructed by support.
- In a Data Guard architecture, backup the primary database to one Recovery Appliance and the standby database to another separate Recovery Appliance. Do not backup a Data Guard primary and the standby to the same Recovery Appliance.
- In a Data Guard architecture, do not send to the same Recovery Appliance backups from the primary and redo logs from the standby, or backups from the standby with redo logs from the primary.
- Dual backup strategies can be complex to maintain for long-term usage and should be avoided if possible. Dual backup strategies are designed for migration from non-Recovery Appliance solutions to an Oracle Zero Data Loss Recovery Appliance solution.
- Use multi section backups: For a Recovery Appliance backup of a bigfile tablespace with an 8K block size, use a `SECTION SIZE` of `64GB`, because it allows for efficient processing in the Recovery Appliance’s flash cache.

  For 8k block size bigfile tablespaces > `16TB`, use `SECTION SIZE` of `128GB`.

  If you have extremely large datafiles (larger than 16TB), please open an SR for guidance.
- Run System Activity Report (SAR) on a daily basis and preferably during the same time of the day.
- Regularly monitor and review Enterprise Manager (EM) notifications for any errors or warnings generated by the Recovery Appliance to proactively identify and address

Copyright © 2014, 2026, Oracle and/or its affiliates.

potential issues before they become critical, ensuring optimal system performance and reliability.

If the Oracle Enterprise Manager is not deployed in your environment, leverage the Recovery Appliance System Activity Report (SAR).

- Perform the `Exachk` health check monthly, thoroughly review the findings, and take corrective action as needed.

  Run `Exachk` before and after patching to ensure a smooth and successful upgrade process. To track changes over time, use the '`diff`' command to compare the monthly `Exachk` reports, allowing for easy identification of new issues or changes in the system's configuration.

- When integrating databases with a newly installed Oracle Zero Data Loss Recovery Appliance, stagger the initial Level 0 backups to avoid overwhelming the system.
- The following changes are considered best practices when working with the Recovery Appliance
  - `CONFIGURE BACKUP OPTIMIZATION ON;`
  - `CONFIGURE CONTROLFILE AUTOBACKUP ON;`
  - `CONFIGURE DEFAULT DEVICE TYPE TO 'SBT_TAPE';`
  - Enable the Block Change Tracking File feature.
  - Enable the Flashback Database feature with a minimum of 60-minute retention to avoid recreation of the standby database following a failover.
- Leverage the Data Guard Broker to configure real-time redo in a Data Guard environment for consistent management with existing broker configuration and settings.
- Maximize storage efficiency and improve backup and restore performance by enabling space-efficient encrypted backups.

  By setting `RA_FORMAT=true` in RMAN channel specification, backups are compressed and encrypted with incremental forever strategy, reducing backup and restore data volume over the network.

  Note that compressed data in the database is not compressed further in the backups.

- For databases with fluctuating workloads or when onboarding new databases to the Oracle Zero Data Loss Recovery Appliance, set the `autotune_reserved_space` parameter to allow the system to dynamically adjust and determine the optimal reserved space value based on the ingest workload, ensuring efficient resource allocation for space management activities.

Copyright © 2014, 2026, Oracle and/or its affiliates.

20

Storage Management for the Recovery Appliance

The Oracle Zero Data Loss Recovery Appliance has manages its storage space for backups with these main controls.

- **Disk Reserved Space**

  `DISK_RESERVED_SPACE` is a database attribute that allows you to manually provision the amount of space needed to hold backups for your database. As long as this is large enough to hold all your backups, the value set by `RECOVER_WINDOW_GOAL` in the protection policy will be safely maintained.

  If space is available on the Recovery Appliance, you may exceed your `DISK_RESERVED_SPACE` needs to help maintain your `RECOVER_WINDOW_GOAL`.

  By managing these value carefully, you can ensure your most important databases always have enough space to meet their goal.

  Less important databases can be limited to ensure more important databases get their needs met. A database that is being limited is trimmed first by reducing the number of days of backups that are being hosted, but if only one complete back exists and this database is still above its `RECOVER_WINDOW_GOAL`. If these conditions are met, then the oldest pieces are simply removed until Recovery Appliance usage is corrected. This will break recovery for that database.

- **Autotuned Reserve Space**

  With `AUTOTUNE_RESERVED_SPACE` set to `YES` in the protection policy, databases in the protection policy are treated equally. The `DISK_RESERVED_SPACE` value for each database is automatically managed. When space runs out on the Recovery Appliance, all databases may lose some backups.

- **Recovery Window Compliance**

  `RECOVER_WINDOW_COMPLIANCE` guarantees the requested window of backups is always maintained. In this case `DISK_RESERVED_SPACE` must be carefully maintained, because if there is insufficient space on the Recovery Appliance to meet `RECOVER_WINDOW_COMPLIANCE` of backups, new backups are rejected until old backups age out of the recover window.

  With `AUTOTUNE_RESERVED_SPACE` and `RECOVER_WINDOW_COMPLIANCE` used together, new backups may be rejected, but only when the whole Recovery Appliance is unable to accomodate the new backups.

Copyright © 2014, 2026, Oracle and/or its affiliates.

21

Running Recovery Appliance Checks

Recovery Appliance checks validate that its components are in a stable and healthy state.

The checks are available through the RACLI utility and can be run together or individually. The Recovery Appliance component checks include:

- **ZDLRA Services** - verifies whether the Recovery Appliance Services (RA Server, DB, CRS) are online.
- **Compute Server Alerts** - checks the compute nodes for `dbmcli` alert history with severity greater than warnings.
- **Storage Server Alerts** - checks the storage cells for `dbmcli` alert history with severity greater than warnings.
- **Active Incidents in the Database** - checks the Recovery Appliance Database for incidents in the database. Can often be bypassed during patching with `'–ignore_incidents'` during the patch appliance steps.
- **Invalid Objects in the Database** - checks the Recovery Appliance database for invalid objects that need to be recompiled.
- **Consistency between the Deployed vs Installed RA Automation RPM** - checks the Recovery Appliance to ensure the deployed RPM vs the installed RPM are consistent.
- **Exadata Image Version Consistency Across All the Hosts** - checks the compute nodes and storage cells to ensure there is only one (1) existing image version for consistency.
- **Init Parameter Validation** - checks the Recovery Appliance database to confirm that `set init` parameters are consistent for a Recovery Appliance configuration.
- **Export Bundle Availability** - checks Recovery Appliance to ensure an export bundle has been successfully taken. In the event of a disaster/crash, the export bundle is used to rebuild the Recovery Appliance to a known working state. The export bundle must be copied to a safe system or location before the Recovery Appliance is rebuilt.
- **Oracle Password Status** - checks to ensure the oracle password has not expired.
- **RASYS User Wallet Status** - checks the validity of the `rasys` wallet. This is required for operations including patching, expansion and upgrade.

racli list check

Use the `racli list check` command to learn the spelling and enabled status of the various checks.

1. From the compute server as `raadmin` group member, run the command.

```
[adminra1@zdlra05 ~]# racli list check
active_incidents
agent_list
appliance_status
cell_alerts
compute_alerts
filesystem_space
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
image_versions
init_parameter
invalid_objects
oracle_password
osb_pieces
ra_catalog
ra_compliance
ra_export
ra_fips
ra_partner
ra_prechecks
ra_quota
ra_selinux
ra_version
ra_vip
rasys_wallets
service_health
session_count
tls_health
[adminra1@zdlra05 ~]#
```

2. List which checks are enabled.

```
[adminra1@zdlra05 ~]# racli list check --status=enabled
active_incidents
appliance_status
cell_alerts
compute_alerts
image_versions
init_parameter
invalid_objects
ra_export
ra_version
[adminra1@zdlra05 ~]#
```

3. List which checks are disabled.

```
[adminra1@zdlra05 ~]# racli list check --status=disabled
ra_prechecks
[adminra1@zdlra05 ~]#racli list check --status=disabled --verbose
ra_prechecks
VERSION=1.0.0.0
GROUP_NAME=DEV
SCRIPT=/opt/oracle.RecoveryAppliance/bin/ra_prechecks.pl
TYPE=system
OPTS=''
ORDER=15
ENABLED=NO
DB_USER=''
[adminra1@zdlra05 ~]#
```

racli run check

Recovery Appliance checks can be run one or more at a time, or all checks that are enabled.

Copyright © 2014, 2026, Oracle and/or its affiliates.

1. From the compute server as `raadmin` group member, run the command.

```
[adminra1@zdlra05 ~]# racli run check --
check_name=active_incidents,invalid_objects
Wed Oct 10 13:53:07 2018: Start: racli run check --
check_name=active_incidents,invalid_objects
HOST: [nnnnnn01.oracle.com]

Created log file scas10adm01.us.oracle.com:/opt/
oracle.RecoveryAppliance/log/racli_run_check_20181010.1353.log
Wed Oct 10 13:53:07 2018: CHECK: Active Incidents - PASS
Wed Oct 10 13:53:09 2018: CHECK: Invalid Objects - PASS
Wed Oct 10 13:53:09 2018: End: racli run check --
check_name=active_incidents,invalid_objects
HOST: [nnnnnn01.oracle.com]
[adminra1@zdlra05 ~]#
```

2. Run all checks that are enabled.

```
[adminra1@zdlra05 ~]# racli run check --all
Wed Oct 10 13:50:28 2018: Start: racli run check --all
HOST: [nnnnnn01.oracle.com]

Created log file scas10adm01.us.oracle.com:/opt/
oracle.RecoveryAppliance/log/racli_run_check_20181010.1350.log

Wed Oct 10 13:50:29 2018: CHECK: RA Services - PASS
Wed Oct 10 13:50:32 2018: CHECK: Compute Node AlertHistory
Wed Oct 10 13:50:32 2018: HOST: [nnnnnn01] - PASS
Wed Oct 10 13:50:32 2018: HOST: [nnnnnn01] - PASS
Wed Oct 10 13:50:43 2018: CHECK: Storage Cell AlertHistory
Wed Oct 10 13:50:43 2018: HOST: [scyyyyyyyadm09] - PASS
Wed Oct 10 13:50:43 2018: HOST: [scyyyyyyyadm05] - PASS
Wed Oct 10 13:50:43 2018: HOST: [scyyyyyyyadm03] - PASS
Wed Oct 10 13:50:43 2018: HOST: [scyyyyyyyadm07] - PASS
Wed Oct 10 13:50:43 2018: HOST: [scyyyyyyyadm01] - PASS
Wed Oct 10 13:50:43 2018: HOST: [scyyyyyyyadm04] - PASS
Wed Oct 10 13:50:43 2018: HOST: [scyyyyyyyadm02] - PASS
Wed Oct 10 13:50:43 2018: HOST: [scyyyyyyyadm06] - PASS
Wed Oct 10 13:50:43 2018: HOST: [scyyyyyyyadm08] - PASS
Wed Oct 10 13:50:44 2018: CHECK: ZDLRA Version
Wed Oct 10 13:50:44 2018: HOST: [scyyyyyyyadm02] - FAIL
Wed Oct 10 13:50:44 2018:
Wed Oct 10 13:50:44 2018: CAUSE:
Wed Oct 10 13:50:44 2018: Unexpected ZDLRA version found.
Wed Oct 10 13:50:44 2018: For more details, see log file:
Wed Oct 10 13:50:44 2018: - /opt/oracle.RecoveryAppliance/log/
racli_check_ra_versions_20181010.1350.log
Wed Oct 10 13:50:44 2018:
Wed Oct 10 13:50:44 2018: HOST: [scyyyyyyyadm01] - FAIL
Wed Oct 10 13:50:44 2018:
Wed Oct 10 13:50:44 2018: CAUSE:
Wed Oct 10 13:50:44 2018: Unexpected ZDLRA version found.
Wed Oct 10 13:50:44 2018: For more details, see log file:
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
Wed Oct 10 13:50:44 2018: - /opt/oracle.RecoveryAppliance/log/
racli_check_ra_versions_20181010.1350.log
Wed Oct 10 13:50:44 2018:
Wed Oct 10 13:50:53 2018: CHECK: Exadata Image Version - PASS
Wed Oct 10 13:50:53 2018: CHECK: Active Incidents - PASS
Wed Oct 10 13:50:56 2018: CHECK: Init Parameters - FAIL
Wed Oct 10 13:50:56 2018:
Wed Oct 10 13:50:56 2018: CAUSE:
Wed Oct 10 13:50:56 2018: Init Parameter Error found
Wed Oct 10 13:50:56 2018: ZDLRA DB Init Parameter Errors:
Wed Oct 10 13:50:56 2018: For more details, see log file:
Wed Oct 10 13:50:56 2018: - /opt/oracle.RecoveryAppliance/log/
racli_check_init_params_20181010.1350.log
Wed Oct 10 13:50:56 2018:
Wed Oct 10 13:50:56 2018: Parameter: _report_capture_cycle_time
Wed Oct 10 13:50:56 2018:
Wed Oct 10 13:50:56 2018: Instance ID: 1
Wed Oct 10 13:50:56 2018: Recomended Value: N/A
Wed Oct 10 13:50:56 2018: Actual Value: 0
Wed Oct 10 13:50:56 2018: Error Text: Init Parameters have non default
value
Wed Oct 10 13:50:56 2018:
Wed Oct 10 13:50:56 2018: Instance ID: 2
Wed Oct 10 13:50:56 2018: Recomended Value: N/A
Wed Oct 10 13:50:56 2018: Actual Value: 0
Wed Oct 10 13:50:56 2018: Error Text: Init Parameters have non default
value
Wed Oct 10 13:50:56 2018:
Wed Oct 10 13:50:56 2018: Please run dbms_ra_adm.update_init_param
Wed Oct 10 13:50:56 2018: in SQL env and bounce database to make them
Wed Oct 10 13:50:56 2018: validate.
Wed Oct 10 13:50:57 2018: CHECK: Invalid Objects - PASS
Wed Oct 10 13:50:58 2018: CHECK: Export Backup - PASS
Wed Oct 10 13:50:58 2018: End: racli run check --all
HOST: [nnnnnn01.oracle.com]
[adminra1@zdlra05 ~]#
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

22

Updating the Recovery Appliance

This section outlines the steps required to install a software update on the Recovery Appliance.

Out-of-place patching is a feature of RA 23.1 (Jan 2025 PSU) for `racli update appliance` and `racli alter appliance` to reduce the downtime required to patch the Recovery Appliance when an update is required. This handles patch steps that do not require the Recovery Appliance to be offline, such as creation of new DB and grid home directories. It requires the directory path to where the gold images for both the DB and GI are stored. Using gold images makes the patch process more consistent and manageable.

The prerequisite for installing update software on the Recovery Appliance is obtaining from Oracle the appropriate update file. This update file is named `<ZIP_NAME_HERE>` in the example below but typically has specific version information in the name.

These steps need to be run as an `admin` or `root` user.

1. Stage the software release by copying the update file (`<ZIP_NAME_HERE>`) to the first compute node of the Recovery Appliance in the `/raacfs/raadmin/tmp/` directory.

   ```
   scp <ZIP_NAME_HERE> <USER>@firstnode:/raacfs/raadmin/tmp/
   ```

2. Stage the Gold image zip files in the `/radump/` directory.

   ```
   scp <GI GOLD IMAGE> <DB GOLD IMAGE> <USER>@firstnode:/radump/
   ```

3. Run the following pre-installation command.

   ```
   <USER>@firstnode # racli run preinstall
   --zdlra_release_file=/raacfs/raadmin/tmp/<ZIP_NAME_HERE>
   ```

   Pay attention to any validations that failed , because they may indicate that the Recovery Appliance is not in an appropriate state to continue.

4. Run `racli alter appliance` to have it complete any *out-of-place* operations that it can without downtime.

   ```
   <USER>@firstnode # racli alter appliance
   [--online updates] --gold_images_path=/radump
   ```

5. Run racli update appliance

   ```
   <USER>@firstnode # racli update appliance [--ignore_incidents]
   ```


Copyright © 2014, 2026, Oracle and/or its affiliates.

> **Note**
>
> If the update fails in the middle and after the identified error is corrected, this command can be rerun and it will continue with the step that failed.
>
> This command replaces `racli patch appliance` and `racli upgrade appliance` and determines whether the update should be a patch or an upgrade. It performs all the prechecks. If the prechecks pass, it proceeds with the patch or upgrade steps.

6. Validate the ZDLRA version that was installed on the Recovery Appliance.

   ```
   <USER>@firstnode # racli version
   ```

7. Validate the ZDLRA services are online.

   ```
   <USER>@firstnode # racli status appliance
   ```

8. Validate the health of the Recovery Appliance.

   ```
   <USER>@firstnode # racli run check --all
   ```

> **Note**
>
> More information about installation and updating the recovery applicance can be found with MOS Note Doc ID KB212582, *Zero Data Loss Recovery Appliance Upgrade and Patching* .

Copyright © 2014, 2026, Oracle and/or its affiliates.

23
Parameter Update Rollback

Allows an update on the parameters to be set back to its previous state.

1. Copy the upgrade file (`zdlra_release.zip`) to the Recovery Appliance compute server into the `/radump/` directory.

   ```
   scp zdlra_release.zip oracle@zdlraadm01:/radump/
   ```

2. As a user in the `raadmin` group, run various checks to verify that the Recovery Appliance is in a healthy and stable state before. Refer to Running Recovery Appliance Checks. As an example:

   ```
   [adminra1@zdlra05 ~]# racli list check --status=enabled
   check_active_incidents
   check_appliance_status
   check_cell_alerts
   check_compute_alerts
   check_image_versions
   check_init_parameter
   check_invalid_objects
   check_ra_export
   check_ra_version
   [adminra1@zdlra05 ~]#
   ```

   ```
   [adminra1@zdlra05 ~]# racli run check --
   check_name=check_active_incidents,check_invalid_objects
   Wed Oct 10 13:53:07 2018: Start: racli run check --
   check_name=check_active_incidents,check_invalid_objects
   HOST: [nnnnnn01.oracle.com]

   Created log file scas10adm01.us.oracle.com:/opt/
   oracle.RecoveryAppliance/log/racli_run_check_20181010.1353.log
   Wed Oct 10 13:53:07 2018: CHECK: Active Incidents - PASS
   Wed Oct 10 13:53:09 2018: CHECK: Invalid Objects - PASS
   Wed Oct 10 13:53:09 2018: End: racli run check --
   check_name=check_active_incidents,check_invalid_objects
   HOST: [nnnnnn01.oracle.com]
   [adminra1@zdlra05 ~]#
   ```

   You can selectively run other checks, or all enabled checks with `racli run check --all`.

   Pay particular attention to any checks that `FAIL`, because they may indicate that the Recovery Appliance is not in an appropriate state to continue with the upgrade operation.

3. On the compute sever, navigate into the `/radump/`directory and unzip the upgrade file.

   ```
   [adminra1@zdlra05 ~]# cd /radump/
   [adminra1@zdlra05 radump]# unzip zdlra_release.zip
   ```


Copyright © 2014, 2026, Oracle and/or its affiliates.

4. As a user in the `raadmin` group, run the following command from the `/radump` directory.

   ```
   [adminra1@zdlra05 radump]# /usr/bin/perl ra_preinstall.pl
   ```

   Pay attention to any validations that failed , because they may indicate that the Recovery Appliance is not in an appropriate state to continue.

5. As a user in the `raadmin` group, run the following command from the `/radump` directory.

   ```
   [adminra1@zdlra05 radump]# /usr/bin/perl ra_preinstall.pl --rollback
   ```

   Pay attention to any validations that failed , because they may indicate that the Recovery Appliance is not in an appropriate state to continue.

6. Validate the ZDLRA version that was installed on the Recovery Appliance.

   ```
   [adminra1@zdlra05 radump]# /opt/oracle.RecoveryAppliance/bin/racli version
   ```

7. Validate the ZDLRA services are online.

   ```
   [adminra1@zdlra05 radump]# /opt/oracle.RecoveryAppliance/bin/racli status
   appliance
   ```

8. Validate the health of the Recovery Appliance.

   ```
   [adminra1@zdlra05 radump]# /opt/oracle.RecoveryAppliance/bin/racli run
   check --all
   ```


Copyright © 2014, 2026, Oracle and/or its affiliates.

24

Migrating Backups to other Recovery Appliances

This chapter explains the migration of backups between Recovery Appliances for maintenance purposes, such as upgrading to newer Recovery Appliances from older Recovery Appliances.

The migration assistant in RACLI is designed to seemlessly transition backups from older Recovery Appliances (source) to newer Recovery Appliances (destination). It clones the users and protection policies from the source and ensures that they are added to the destination Recovery Appliance. The migration assistant establishes on the destination Recovery Appliance the same configuration that exists on the source. This includes the replication server (if it was on the source), protection policies, VPC users, OKV endpoints configuration on the Recovery Appliance for copy-to-cloud, and Oracle Secure Backup (OSB) for copy-to-tape. Finally, all of the protected databases from the source Recovery Appliance are copied onto the destination Recovery Appliance.

> **Note**
>
> Oracle Secure Backup (OSB) 19.1 desupport was announced on May 1, 2026 with premier support end date on September 30, 2027. See My Oracle Support note PNEWS3035 for more information. Please refer to Lifetime Support Policy: Oracle Technology Products for updated support timeframes.
>
> Oracle recommends that database backups to cloud take advantage of Oracle Database Zero Data Loss Cloud Protect, which uses Oracle Zero Data Loss Autonomous Recovery Service in OCI. Cloud Protect offers an efficient incremental forever backup strategy, real-time transaction protection, logically air-gapped immutable backups, and fast, point-in-time recovery.
>
> For customers who require database backups to Amazon S3, Oracle Database Cloud Backup for Amazon S3 may be used. This offering is similar to Oracle Secure Backup Cloud Module in that it is fully integrated with Recovery Manager (RMAN) and enables you to back up your Oracle database to Amazon S3. You can exchange your Oracle Secure Backup license for an Oracle Database Cloud Backup for Amazon S3 license on a 1:1 basis. See Oracle Database Cloud Backup for Amazon S3 for more information.
>
> As with Oracle Secure Backup Cloud Module, Oracle Database Cloud Backup for Amazon S3 can be used free of charge on Oracle Zero Data Loss Recovery Appliance (ZDLRA) for the sole purpose of backing up data to and restoring data from S3-compatible storage devices that are in the same campus as ZDLRA.

The migration assistant performs several pre-check operations.

- Verifies the RACLI versions match between source and destination.
- Verifies the TLS settings match between source and destination.
- Verifies the storage location size of the destination matches or is greater than the source's storage size.

Copyright © 2014, 2026, Oracle and/or its affiliates.

- Verifies the new OKV jar file is downloaded and staged on the destination Recovery Appliance. It verifies both Recovery Appliances share the same wallet. This means client-to-client symmetric encryption using a single, shared key matches between souce and destination Recovery Appliances. During migration, it performs OKV endpoint enrollment on the destination Recovery Appliance, which allows copy-to-cloud operations that were performed on the source to work in the same way on the destination.
- Verifies the status of the source Recovery Appliance copy-to-tape operations are disabled or paused.
- Verifies no conflicts in database names from the source Recovery Appliance to databases that already exist on the destination. If the target Recovery Appliance already has users or policies named the same as on the source, they are skipped during migration. Copy-to-tape library names are not part of the migration process.

After the pre-checks, the migration assistant collects meta data from the source Recovery Appliance, builds a migration bundle, and creates a migration server. On the destination Recovery Appliance, the migration assistant clones metadata of the users, protection policies, and protected databases. If the source Recovery Appliance had a replication server, this replication server is created on the destination Recovery Appliance. It adds a policy to the migration server, where a `COPYALL_BACKUPS` is initiated. The `COPYALL_BACKUPS` replicates all of the existing backups for the database to the destination Recovery Appliance, including backups that might be expired but not yet purged from the source Recovery Appliance. The source replicates the backups in controlled and orderly fashion, while the destination, being `COPYALL_BACKUPS` aware, ingests and maintains the backups in the correct order.

> **Note**
>
> The migration process from a source to a destination Recovery Appliance can take an extended time period to complete due to the volume of backup data that needs to be copied through available network bandwidth.

The migration assistant checks and confirms the `COPYALL_BACKUPS` status. Upon completion, it removes and deletes the migration server from the source Recovery Appliance.

24.1 Configuring Migration between Recovery Appliances Using RACLI

Partnership Overview

The migration partnership links two different Recovery Appliances (RAs) and allows for remote management with a non-root user. The partnership:

- Establishs a one-to-one relationship between two Recovery Appliances and is removed after migration completes.
- Creates an internal "partner" OS user on the source Recovery Appliance that has SSH access to the destination in order to perform specific RACLI commands.
- Establishes two-way communication, therefore the command only needs to be run on one Recovery Appliance.

The migration operations should be performed using the RACLI commands.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 24-1 Principal RACLI Commands for Migration

Program Unit	Description
`racli add ra_partner --target_host=FQDN [--partner_user=NAME] [--partner_uid=UID] [--admin_user=VALUE] [--admin_key=VALUE]`	Establishes a two-way connection between a source and destination Recovery Appliance. It creates a non-root 'partner_user' with limited privileges, and sets up SSH key-based 2-way authentication.
`racli list ra_partner`	Shows all of the information on the partner Recovery Appliance including its host name.
`racli alter ra_partner { --target_host=FQDN } [--partner_user=VALUE] [--partner_dir=VALUE] [--admin_user=VALUE] [--admin_key=VALUE]`	Updates the partner's SSH key and other details on source Recovery Appliance to share with destination.
`racli remove ra_partner --target_host=FQDN [--partner_user=VALUE] [--partner_dir=VALUE] [--admin_user=VALUE] [--admin_key=VALUE]`	Run on source Recovery Appliance to remove objects and configuration created to support the partner relationship between source and destination Recovery Appliance used during migration.
`racli begin migration`	Begins migration from source Recovery Appliance to destination Recovery Appliance.
`racli status migration`	Returns status of migration server and `COPYALL_BACKUPS` backup state.
`racli end migration`	Cleans up migration server between source and the destination Recovery Appliances.

Migration Notes

- **Extra Caution with Database Operations:** Be extra cautious when adding, deleting or renaming databases while the database migration is still in progress. Ensure that the operations are carried out on both partner Recovery Appliances.
- **Storage Space:** If the source Recovery Appliance (RA1) is migrating to a single destination Recovery Appliance (RA2), the RA2 must have a larger storage location size than the source RA1 to pass the precheck. This ensures there is enough space to copy all the data from the source RA1.
- **Replication Server on Source:** Only one replication server is allowed to be associated with the source Recovery Appliance (RA1) before migration. If the source RA1 had originally more than one replication servers, the additional replication servers need to be removed; migration to RA2 will only set up one replication server.
- **Single migrations without cloning replication:** If the source Recovery Appliance had a replication server configured, the migration process establishes this on the destination Recovery Appliance.
- **Copy-to-Cloud:** If the source Recovery Appliance (RA1) for a migration has a copy-to-cloud setup including master keys (checked via `racli list okv_util`), then this OKV must be registered on the destination Recovery Appliance (RA2) through RA2's web console prior to migration. [Refer to Installing the OKV Client Software.]

Copyright © 2014, 2026, Oracle and/or its affiliates.

The master key must be downloaded to each node of the destination host. If this is not done, the precheck may fail indicating that the master key is found on the local Recovery Appliance but not on the destination Recovery Appliance.

- **Migration of Protection Policies:** The migration procedure is intended to migrate all protection policies and their associated databases to the destination Recovery Appliance in order to be able to decommission the source Recovery Appliance. You can also migrate *"by protection policy"*, meaning that only the databases associated with the named protection policy are migrated. This is specified using "`racli begin migration`" with the argument `--protection_policy_name=<POLICY_NAME>`.
- **Missing Partnerships:** This causes failures during the precheck. For instance, if the source Recovery Appliance (RA1) was replicating with Recovery Appliance RA3 and the source RA1 is now migratiing to Recovery Appliance RA2, a partnership between RA2 and RA3 is necessary.

  ```
  racli add ra_partner --target_host=RA3_full_name --partner_user=RA2_RA3
  ```

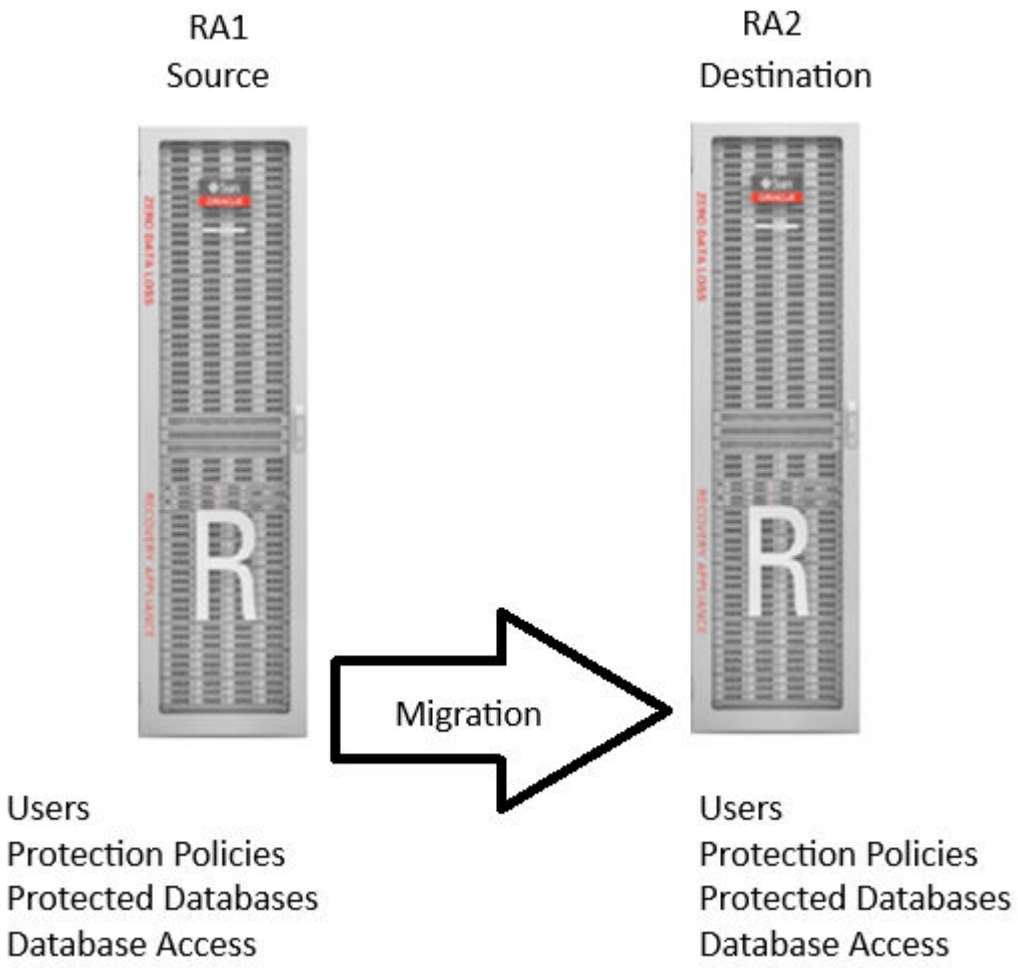
If the source Recovery Appliance (RA1) had existing replication server to a Recovery Appliance (RA3), the migration assistant creates the replication server and adds protection policies for replication between the destination Recovery Appliance (RA2) and replication Recovery Appliance (RA3).

Use Case 1: Migrating from RA1 and RA2

When migrating backups from a single Recovery Appliance to another, follow these steps.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Figure 24-1 Migration between two Recovery Appliances



1. On RA1 (source), establish a partnership to RA2 (destination / target).

   ```
   racli add ra_partner  --target_host=<RA2>
   ```

2. On RA1, start the migration process.

   ```
   racli begin migration  --target_host=<RA2>
   ```

   > **Note**
   >
   > The migration process creates, adds, and uses the migration server. Upon successful completion, it deletes and removes the migration server and cleans up any migration-related processes.

3. Review and fix any pre-check errors. Then rerun `racli begin migration`.
4. Database administrator can change the client (database) backup to point to RA2.

Copyright © 2014, 2026, Oracle and/or its affiliates.

5. On RA1, check on the status of the migration process.

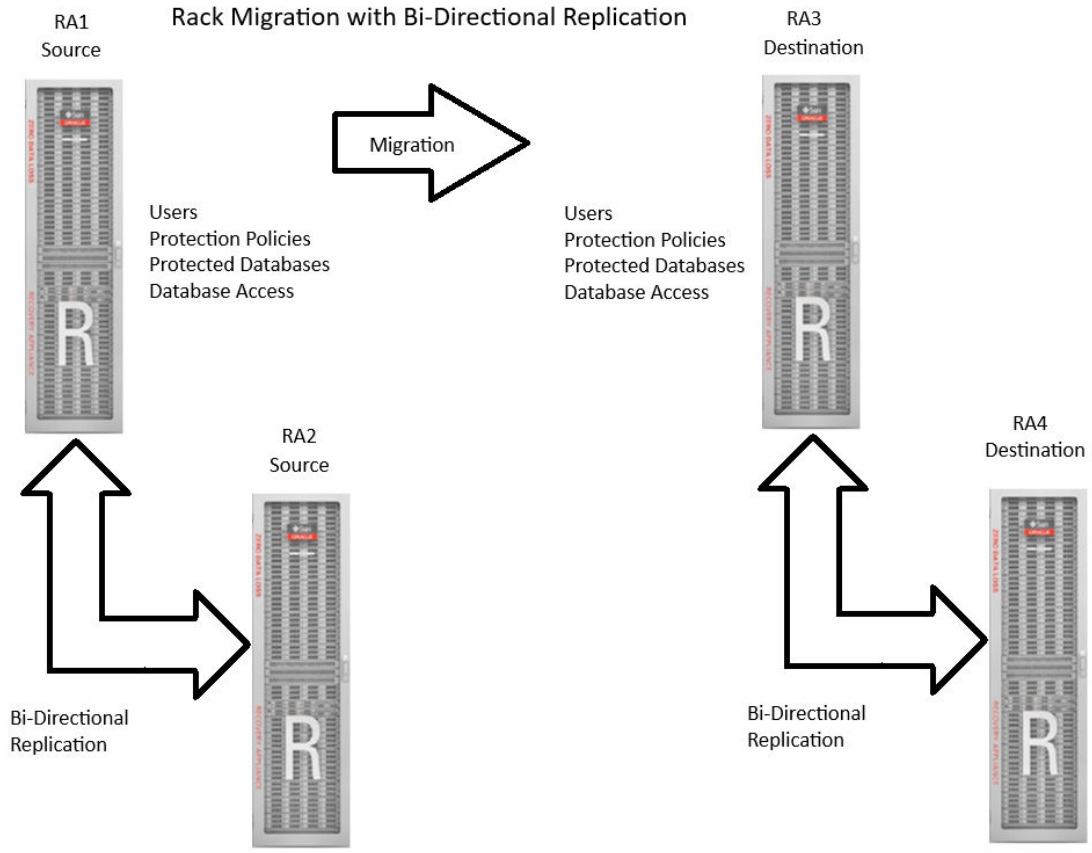
```
racli status migration
```

6. When the status shows a `COMPLETE` state, end migration and cleanup migration server.

```
racli end migration --target_host=<RA2>
```

Use Case 2: Migrating from a bidirectional replication pair RA1 & RA2 to another replication pair RA3 and RA4

Figure 24-2 Rack Migration with Bi-Directional Replication



Procedures:

1. On RA1 (source), setup RA partnerships.

   Setup RA partnership between RA1 and RA3.

```
racli add ra_partner --target_host=RA3_full_name
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

On RA1 (source), setup RA partnership between RA1 and RA4. This allows RA1 to partner with RA3 and RA4 and serve as the control center for the overall migration process.

```
racli add ra_partner --target_host=RA4_full_name
```

On RA2 (source, because this is the original replication), setup RA partnership between RA2 and RA4.

```
racli add ra_partner --target_host=RA4_full_name
```

2. Start the migration process.

```
racli begin migration  --target_host=<RA2>
```

> **Note**
>
> The migration process creates, adds, and uses the migration server. Upon successful completion, it deletes and removes the migration server and cleans up any migration-related processes.

3. Database administrator can change the client (database) backup to now point to RA3 and RA4 (assuming bi-directional replication).
4. On RA1 (source), regularly check on the status of the migration process.

```
racli status migration
```

5. When the status shows `COMPLETE`, you can end migration, which cleans up all migration process components, including the migration server.

```
racli end migration --target_host=<RA2>
```

Use Case 3: Migrating by protection policy from RA1 and RA2

This use case is similar to Use Case 1, except that a protection policy is provided when starting the migration. As a result, only the databases associated with the named protection policy are migrated from the source RA1 to the target RA2 Recovery Appliance.

1. On RA1 (source), establish a partnership to RA2 (destination / target).

```
racli add ra_partner  --target_host=<RA2>
```

2. On RA1, specify the protection policy when you start the migration process.

```
racli begin migration  --target_host=<RA2> --
protection_policy_name=<POLICY_NAME>
```

All databases associated with the named protection policy are migrated.

Copyright © 2014, 2026, Oracle and/or its affiliates.

> **Note**
>
> The migration process creates, adds, and uses the migration server. Upon successful completion, it deletes and removes the migration server and cleans up any migration-related processes.

3. Review and fix any pre-check errors. Then rerun `racli begin migration`.
4. On RA1, check on the status of the migration process.

   ```
   racli status migration
   ```

5. When the status shows a `COMPLETE` state, you end the migration and cleanup migration server.

   ```
   racli end migration --target_host=<RA2>
   ```


Copyright © 2014, 2026, Oracle and/or its affiliates.

Part III
Recovery Appliance Reference

Part II contains the following chapters:

- DBMS_RA Package Reference
- Recovery Appliance View Reference
- rastat Utility Reference
- Recovery Appliance Error Message Reference

Copyright © 2014, 2026, Oracle and/or its affiliates.

25

DBMS_RA Package Reference

This chapter provides details on the DBMS_RA PL/SQL package. You use DBMS_RA subprograms to perform all Recovery Appliance administration functions.

One DBMS_RA procedure may execute at a given time. With the exception of ABORT_RECOVERY_APPLIANCE, attempts to run multiple DBMS_RA procedure in different sessions at the same time fail with an appropriate message.

The following table summarizes the available DBMS_RA subprograms.

Table 25-1 DBMS_RS Package Subprograms

Subprogram	Description
ABORT	Synonymous with ABORT_RECOVERY_APPLIANCE.
ABORT_RECOVERY_APPLIANCE	This procedure shuts down the Recovery Appliance without waiting for in-progress operations to complete.
ADD_DB	This procedure adds the specified database to the Recovery Appliance, and assigns a protection policy to the database. This procedure enables a non-protected database to attain the status of a protected database.
ADD_REPLICATION_SERVER	This procedure adds the specified replication server configuration to the specified protection policy. After the operation succeeds, the Recovery Appliance replicates backups of databases protected by this policy to the downstream Recovery Appliance.
CONFIG	This procedure updates a value in the config table. Do not perform parameter changes unless so instructed by Oracle Support for ZDLRA.
COPY_BACKUP	This procedure copies one or more backup pieces from the Recovery Appliance to a user-specified disk or SBT destination. The Recovery Appliance copies all backup pieces matching the specified tag to the location specified with the format and template_name parameters.
COPY_BACKUP_PIECE	This procedure copies a single backup piece from the Recovery Appliance to a user-specified disk or SBT destination.
CREATE_ARCHIVAL_BACKUP	This procedure copies all backup pieces from Recovery Appliance as restricted by user inputs to TAPE with the ability to recover the protected database to a user specified point described by restore_until_scn or restore_until_time.
CREATE_POLLING_POLICY	This procedure creates a backup polling policy.
CREATE_PROTECTION_POLICY	This procedure creates a protection policy.
CREATE_RESTORE_POINT GUID-0A104488-C9F7-4339-9352-CC17C5402E00	This procedure defines a named restore point on the Recovery Appliance that can be used to create an archival backup.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 25-1 (Cont.) DBMS_RS Package Subprograms

Subprogram	Description
CREATE_REPLICATION_SERVER	This procedure defines a configuration for a downstream Recovery Appliance that forms part of a Recovery Appliance replication scheme.
CREATE_SBT_ATTRIBUTE_SET	This procedure creates an SBT attribute set that SBT jobs can use.
CREATE_SBT_JOB_TEMPLATE	This procedure creates an SBT job that describes how the Recovery Appliance chooses backups for copying to tape. This form of this overloaded procedure applies to backups for all protected databases assigned to the specified protection policy.
CREATE_SBT_JOB_TEMPLATE	This procedure creates a new SBT backup job. The job describes how the Recovery Appliance chooses backups for copying to tape/cloud. This form of this overloaded procedure applies to backups for a single protected database only, whereas the previous form applies to backups of all databases assigned to a specific protection policy. With the exception of this difference, this procedure and its parameters are identical to the alternative form of this procedure.
CREATE_SBT_LIBRARY	This procedure creates metadata describing an installed media management software library. The Recovery Appliance uses the specified library to copy backups from internal storage either to tape or to other tertiary storage supported by this media manager.
DELETE_DB	This procedure removes the specified protected database from the Recovery Appliance. The Recovery Appliance deletes all metadata and backups associated with this database, both from disk and SBT. Backups on tape are not affected. Regarding replicated and cloud pieces: the meta data is deleted from the Recovery Appliance doing the delete, but the data is not deleted from the downstream or cloud storage.
DELETE_POLLING_POLICY	This procedure deletes the specified backup polling policy.
DELETE_PROTECTION_POLICY	This procedure deletes the specified protection policy.
DELETE_REPLICATION_SERVER	This procedure deletes a replication server configuration. The Recovery Appliance removes all metadata relating to the downstream Recovery Appliance.
DELETE_SBT_ATTRIBUTE_SET	This procedure deletes the specified SBT attribute set.
DELETE_SBT_JOB_TEMPLATE	This procedure deletes the specified SBT job template.
DELETE_SBT_LIBRARY	This procedure deletes the metadata describing the specified SBT library.
ESTIMATE_SPACE	This procedure estimates the amount of storage in GB required for recovery of a given database and a desired recovery window.
GET_REDO_TRANSPORT_LAG	Returns the current redo transport lag on the specified database.
GRANT_DB_ACCESS	This procedure grants the necessary privileges to the specified recovery Appliance user account to enable this account to back up, restore, and access recovery catalog metadata for the specified protected database.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 25-1 (Cont.) DBMS_RS Package Subprograms

Subprogram	Description
KEY_REKEY	This procedure rekeys encryption keys for all databases with existing encryption keys.
KEY_REKEY	This procedure rekeys encryption keys for the specified database with an existing encryption key.
KEY_REKEY	This procedure rekeys encryption keys for all databases with existing encryption keys in the specified protection_policy
MIGRATE_TAPE_BACKUP	This procedure makes pre-migration tape backups accessible to the Recovery Appliance through the specified SBT library. You must first import metadata about the tape backups into the Recovery Appliance catalog using the RMAN `IMPORT CATALOG` command.
MOVE_BACKUP	This procedure moves one or more long-term archival backup pieces from the Recovery Appliance to a user-specified disk or SBT destination.
MOVE_BACKUP_PIECE	This procedure moves a single long-term archival backup piece from the Recovery Appliance to a user-specified disk or SBT destination.
PAUSE_REPLICATION_DATABASE	This procedure pauses replication for the specified database with all associated replication servers. If `replication_server_name` is specified, replication for the one database/one replication server is paused.
PAUSE_REPLICATION_SERVER	This procedure pauses replication to the specified downstream Recovery Appliance.
PAUSE_SBT_LIBRARY	This procedure pauses the specified SBT library. The Recovery Appliance allows in-progress copies of backup pieces to complete. However, if backup pieces were queued for copy through this SBT library but not yet copied, then the Recovery Appliance holds them until you resume the SBT library. No new SBT jobs that run against this library can execute until you resume the library (RESUME_SBT_LIBRARY).
POPULATE_BACKUP_PIECE	This procedure pushes the specified backup piece into the delta store.
QUEUE_SBT_BACKUP_TASK	This procedure queues the backup pieces selected by the specified SBT job template for copying to tape. Typically, a scheduling utility such as Oracle Scheduler calls this procedure.
REMOVE_REPLICATION_SERVER	This procedure removes the specified replication server configuration from the specified protection policy. After the operation succeeds, the Recovery Appliance no longer replicates backups of databases protected by this policy to the downstream Recovery Appliance.
RENAME_DB	This procedure changes the name of the specified protected database in the Recovery Appliance metadata.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 25-1 (Cont.) DBMS_RS Package Subprograms

Subprogram	Description
RESET_ERROR	This procedure modifies the specified set of incident log entries to have the status RESET. It takes multiple optional input parameters to allow bulk resetting of errors. When two or more input parameters are specified in a single RESET_ERROR call, only records that match all the input parameters specified together are RESET. Errors marked in this fashion do not cause Oracle Enterprise Manager to raise alerts. If the Recovery Appliance determines that the problem is still occurring, then errors that have been reset change to ACTIVE status. The primary use of this API is to reset the error status for nonrecurring errors, such as transient media failures.
RESUME_DB	This procedure restores a suspended database to normal operation. Only suspended databases may be resumed.
RESUME_REPLICATION_DATABASE	This procedure resumes replication for the specified database after a previous call to `pause_replication_database`.
RESUME_REPLICATION_SERVER	This procedure resumes replication to the specified downstream Recovery Appliance, after a previous call to PAUSE_REPLICATION_SERVER.
RESUME_SBT_LIBRARY	This procedure resumes a paused SBT library.
REVOKE_DB_ACCESS	This procedure revokes privileges on one protected database from the specified Recovery Appliance user account.
SET_SYSTEM_DESCRIPTION	This procedure sets a descriptive name for users to apply to their Recovery Appliance. The name provided here will be seen in the RA_SERVER view.
SHUTDOWN	Synonymous with SHUTDOWN_RECOVERY_APPLIANCE.
SHUTDOWN_RECOVERY_APPLIANCE	This procedure performs a clean shutdown of the Recovery Appliance.
STARTUP	Synonymous with STARTUP_RECOVERY_APPLIANCE.
STARTUP_RECOVERY_APPLIANCE	This procedure starts the Recovery Appliance after it has been shut down or terminated.
SUSPEND_DB	This procedure deletes all local disk backups associated with this database from the Recovery Appliance. Backups on tape, in the cloud, or replicated to other Recovery Appliances are not affected.
UPDATE_ARCHIVAL_BACKUP_KEEP	This procedure makes updates the retention time of archival backup with the specified `keep_until_time`. Archival backup is identified by user specified `restore_tag` and `restore_point`.
UPDATE_DB	This procedure changes the attributes that are assigned to the specified protected database.
UPDATE_POLLING_POLICY	This procedure modifies the parameters for an existing backup polling policy.
UPDATE_PROTECTION_POLICY	This procedure modifies the parameters for an existing protection policy.
UPDATE_REPLICATION_SERVER	This procedure changes the settings for a replication server configuration.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 25-1 (Cont.) DBMS_RS Package Subprograms

Subprogram	Description
UPDATE_SBT_ATTRIBUTE_SET	This procedure updates the parameters for the specified SBT attribute set.
UPDATE_SBT_JOB_TEMPLATE	This procedure updates the parameters for the specified SBT job.
UPDATE_SBT_LIBRARY	This procedure modifies the parameters for the specified SBT library.

25.1 ABORT

Synonymous with ABORT_RECOVERY_APPLIANCE.

Syntax

```
PROCEDURE abort(
  comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-2 ABORT Parameters

Parameter	Description
comments	Optional user supplied comment describing reason for executing this command.

25.2 ABORT_RECOVERY_APPLIANCE

This procedure shuts down the Recovery Appliance without waiting for in-progress operations to complete.

When you use this procedure, the Recovery Appliance terminates backup, restore, and background operations (such as validations, data moves, and copy-to-tape jobs) that have started but not completed. When the Recovery Appliance restarts, it automatically resumes or restarts backup operations. You must manually restart any terminated backup and restore operations. To perform a clean shutdown, use SHUTDOWN_RECOVERY_APPLIANCE.

Syntax

```
PROCEDURE abort_recovery_appliance(
  comments IN VARCHAR2 DEFAULT NULL);
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

Parameters

Table 25-3 ABORT_RECOVERY_APPLIANCE Parameters

Parameter	Description
comments	Optional user supplied comment describing reason for executing this command.

25.3 ADD_DB

This procedure adds the specified database to the Recovery Appliance, and assigns a protection policy to the database. This procedure enables a non-protected database to attain the status of a protected database.

Enrolling a database with the Recovery Appliance involves:

adding the protected database with ADD_DB.

granting access to this database to a Recovery Appliance user account (GRANT_DB_ACCESS).

registering this database in the virtual private catalog (RMAN REGISTER DATABASE command). The protected database must be enrolled before Recovery Appliance can process backup and restore operations.

You cannot use this procedure to add additional databases in a physical standby configuration. Such databases will be automatically recognized as they perform recovery catalog resynchronizations.

This procedure adds the specified database to the Recovery Appliance, and assigns a protection policy to the database. This procedure enables a non-protected database to attain the status of a protected database.

Enrolling a database with the Recovery Appliance involves:

adding the protected database with ADD_DB.

granting access to this database to a Recovery Appliance user account (GRANT_DB_ACCESS).

registering this database in the virtual private catalog (RMAN REGISTER DATABASE command). The protected database must be enrolled before Recovery Appliance can process backup and restore operations.

You cannot use this procedure to add additional databases in a physical standby configuration. Such databases will be automatically recognized as they perform recovery catalog resynchronizations.

Syntax

```
PROCEDURE add_db (
   db_unique_name IN VARCHAR2,
   protection_policy_name IN VARCHAR2,
   reserved_space IN VARCHAR2 DEFAULT NULL,
   comments IN VARCHAR2 DEFAULT NULL);
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

Parameters

Table 25-4 ADD_DB Parameters

Parameter	Description
db_unique_name	The DB_UNIQUE_NAME of the database to add.
protection_policy_name	The name of the protection policy to assign to the database. The protection policy must exist.
reserved_space	The amount of disk space that is guaranteed to be available for the protected database. The format of this value is a character string that must contain a number consisting only of the characters 0-9, followed optionally by one of the following unit specifiers: K: Kilobytes M: Megabytes G: Gigabytes T: Terabytes P: Petabytes If no unit is specified, then the Recovery Appliance interprets the value as a number of bytes. reserved_space may be specified as a NULL if the controlling protection policy supports the autotune_reserved_space feature.
comments	Optional user supplied comment describing reason for executing this command.

25.4 ADD_REPLICATION_SERVER

This procedure adds the specified replication server configuration to the specified protection policy. After the operation succeeds, the Recovery Appliance replicates backups of databases protected by this policy to the downstream Recovery Appliance.

See CREATE_REPLICATION_SERVER.

Syntax

```
PROCEDURE add_replication_server (
   replication_server_name IN VARCHAR2,
   protection_policy_name IN VARCHAR2
   skip_initial_replication IN BOOLEAN DEFAULT FALSE,
   read_only IN BOOLEAN DEFAULT FALSE,
   request_only IN BOOLEAN DEFAULT FALSE
   copyall_backups IN BOOLEAN DEFAULT FALSE,
   comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-5 ADD_REPLICATION_SERVER Parameters

Parameter	Description
replication_server_name	The name of the replication server configuration to associate with the protection policy.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 25-5 (Cont.) ADD_REPLICATION_SERVER Parameters

Parameter	Description
protection_policy_name	The name of the protection policy to associate with the replication server configuration.
skip_initial_replication	If set to TRUE, initial replication is skipped.
read_only	The setting that controls replication behavior to a downstream Recovery Appliance. If TRUE, then the Recovery Appliance treats the downstream replication server as a read only device. Specifically, backups are not replicated to the downstream replication server. Backups that exist on the downstream are restorable through the upstream Recovery Appliance. If FALSE or not specified, then the Recovery Appliance does the normal initial replication.
request_only	When adding a replication server to a protection polity with request_only set to TRUE, backups are not replicated to the downstream replication server. At startup time of the local Recovery Appliance, a calculation is made to determine which backups exist on the remote Recover Appliance and not locally. Those backups that exist remotely are requested to be sent from the downstream to the local Recovery Appliance. This feature requires that the two Recovery Appliances are paired with each other. If FALSE or not specified, the Recovery Appliance does normal replication.
copyall_backups	When adding a replication server to a protection policy with copyall_backups set to TRUE, all backups for each of the databases are chosen for initial replication to the downstream Recovery Appliance. If set to FALSE the most recent level 0 including archivelogs and control files are replicated.
comments	Optional user supplied comment describing reason for executing this command.

25.5 CONFIG

This procedure updates a value in the config table.

> **Note**
>
> Do not perform parameter changes for the Recovery Appliance unless so instructed by Oracle Support.

Changes made to the config table are tracked, as well as default values, which are the "best values" that the Recovery Appliance is shipped with.

Syntax

```
PROCEDURE config(
   p_name VARCHAR2,
   p_value VARCHAR2,
   comments IN VARCHAR2 DEFAULT NULL);
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

Parameters

Table 25-6 CONFIG Parameters

Parameter	Description
p_name	The parameter to update. Possible parameters are: check_files_days The frequency with which the Recovery Appliance runs the metadata consistency check in the background. The default frequency is 7 days. crosscheck_db_days The frequency with which the Recovery Appliance performs crosschecks of the recovery catalog to reflect actions in tape libraries or downstream Recovery Appliances. The default frequency is 1 day. optimize_chunks_days The frequency with which the Recovery Appliance performs background re-ordering of blocks in the delta store to reduce disk reads required for restore operations. The default frequency is 7 days. group_log_max_count These are the maximum number of archive logs that are grouped into a single backup before being written to tape. The default is 1, which is equivalent to turning it off. group_log_backup_size_gb This is the maximum size allowed for an archive log backup created with archive log grouping. the default is 256 GB. validate_metadata_days The frequency with which the Recovery Appliance performs background metadata validation. The default frequency is 7 days. validate_db_days The frequency with which the Recovery Appliance performs background validation of backup pieces. The default frequency is 7 days. percent_late_for_warning The percent threshold at which the Recovery Appliance posts warnings for incomplete background operations. For example, if validate_db_days is 7 and the percent_late_for_warning is 50, then the Recovery Appliance records a warning in the incident log when a database has gone 10.5 (or 7 + ((50/100)*7)) days without being validated. The default is 100 percent. network_chunksize The message size that the Recovery Appliance uses for transferring backups between itself and protected databases. It is also used for replication. The default is 128 MB. All protected databases use this value to determine the unit size in which to send or read backups. For example, if a protected database backup is 1 TB, then the SBT library sends the backup data to the Recovery Appliance in units of network_chunksize. This technique is an optimization that enables the Recovery Appliance to restart the data transfer faster if a failure occurs.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 25-6 (Cont.) CONFIG Parameters

Parameter	Description
p_value	The new value for the parameter.
comments	Optional user supplied comment describing reason for executing this command.

25.6 COPY_BACKUP

This procedure copies one or more backup pieces from the Recovery Appliance to a SBT destination. The Recovery Appliance copies all backup pieces matching the specified tag to the location specified with the format and template_name parameters.

Syntax

```
PROCEDURE copy_backup (
   tag IN VARCHAR2,
   format IN VARCHAR2,
   template_name IN VARCHAR2,
   compression_algorithm IN VARCHAR2 DEFAULT NULL,
   encryption_algorithm IN VARCHAR2 DEFAULT NULL,
   comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-7 COPY_BACKUP Parameters

Parameter	Description
tag	The tag of the backups to copy. The Recovery Appliance copies all backups matching this tag.
format	The naming format of the backup pieces to create. This parameter follows the same rules as the RMAN FORMAT parameter. If not specified, default format is RA_SBT_%d_%I_<SBT_job_template_key>_%U_<bs_key>.
template_name	The name of the SBT job library template. The Recovery Appliance copies the backup piece to tape (or cloud), using the media pool referenced in the SBT template name as the copy destination.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 25-7 (Cont.) COPY_BACKUP Parameters

Parameter	Description
`compression_algorithm`	Specifies the compression algorithm. If `compression_algorithm` is specified, it will override the compression algorithm defined in `template_name` for this single operation. If `template_name` is NULL, it defines the compression algorithm for this operation. `BASIC`: Good compression ratios with potentially lower speed than `MEDIUM`. `LOW`: Optimized for speed with potentially lower compression raios than `BASIC`. `MEDIUM`: Recommended for most environments. Good combination of compression ratios and speed. `HIGH`: Best suited for operations over slower networks where the limiting factor is maximum network throughput. Provides the highest level of compression with the most negative impact on CPU performance. `OFF`: No compression. `NULL`: (default) indicates that the algorithm defined in the SBT job template should be used.
`encryption_algorithm`	Specifies the encryption algorithm If `encryption_algorithm` is specified, it will override the encryption algorithm defined in `template_name` for this single operation. If `template_name` is NULL, it defines the encryption algorithm for this operation. Valid values are 'AES128', 'AES192', 'AES256', 'OFF', 'CLIENT', or the constant equivalents ENC_OFF, ENC_AES128, ENC_AES192, ENC_AES256, ENC_CLIENT. **Note** A value of `CLIENT` or `ENC_CLIENT` requires the client to generate encrypted backups. Failure to do so will result in cloud backup job failures on the Recovery Appliance.
`comments`	Optional user supplied comment describing reason for executing this command.

25.7 COPY_BACKUP_PIECE

This procedure copies a single backup piece from the Recovery Appliance to a SBT destination.

Syntax

```
PROCEDURE copy_backup_piece (
   bp_key IN NUMBER,
   format IN VARCHAR2,
   template_name IN VARCHAR2,
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
  compression_algorithm IN VARCHAR2 DEFAULT NULL,
  encryption_algorithm IN VARCHAR2 DEFAULT NULL,
  comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-8 COPY_BACKUP_PIECE Parameters

Parameter	Description
bp_key	The unique key of the backup piece to copy. Obtain this key from the RC_BACKUP_PIECE view.
format	The naming format of the backup piece to create. This parameter follows the same rules as the RMAN FORMAT parameter. If not specified, default format is RA_SBT_%d_%I_<SBT_job_template_key>_%U_<bs_key>.
template_name	The name of the SBT job library template. The Recovery Appliance copies the backup piece to tape, using the media pool referenced in the SBT template name as the copy destination.
compression_algorithm	Specifies the compression algorithm. If compression_algorithm is specified, it will override the compression algorithm defined in template_name for this single operation. If template_name is NULL, it defines the compression algorithm for this operation. BASIC: Good compression ratios with potentially lower speed than MEDIUM. LOW: Optimized for speed with potentially lower compression raios than BASIC. MEDIUM: Recommended for most environments. Good combination of compression ratios and speed. HIGH: Best suited for operations over slower networks where the limiting factor is maximum network throughput. Provides the highest level of compression with the most negative impact on CPU performance. OFF: No compression. NULL: (default) indicates that the algorithm defined in the SBT job template should be used.
encryption_algorithm	Specifies the encryption algorithm If encryption_algorithm is specified, it will override the encryption algorithm defined in template_name for this single operation. If template_name is NULL, it defines the encryption algorithm for this operation. Valid value are 'AES128', 'AES192', 'AES256', 'OFF' or the constant equivalents ENC_OFF, ENC_AES128, ENC_AES192, ENC_AES256.

> **Note**
>
> A value of CLIENT or ENC_CLIENT requires the client to generate encrypted backups. Failure to do so will result in cloud backup job failures on the Recovery Appliance.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 25-8 (Cont.) COPY_BACKUP_PIECE Parameters

Parameter	Description
comments	Optional user supplied comment describing reason for executing this command.

25.8 CREATE_ARCHIVAL_BACKUP

This procedure copies all backup pieces from Recovery Appliance as restricted by user inputs to TAPE with the ability to recover the protected database to a user specified point described by restore_until_scn, restore_until_time, or restore_point.

Backups created on TAPE using this API are KEEP backups and preserved until user specified keep_until_time.

Archival backups are validated using restore_point and restore_tag. If restore_point is not specified, it is generated internally. If restore_tag is not specified, tag values for archival backups will be the same as restore_point name.

Format for internally generated restore_point name is: <KEEP_BACKUP_><yyyyMMddHH24miSS>

> **Note**
>
> The database must have archivelog mode turned on prior to trying to create an archival backup. This command requires the archive logs to properly compute the necessary files to create a complete consistent backup for archival purposes.

This API has the following restrictions for input options:

- If a restore_point is specified and it doesn't exist, then a new restore_point is created with the specified restore_point name. In this case, an additional input can be specified: either restore_until_scn or restore_until_time but not both.
- If a restore_point is specified and that restore_point exists, then the user cannot specify any additional input parameters.
- If the specified restore_until_time is not within the low_time and the high_time for the database, then this API returns an error
- The restore_until_time is the time up to which backups are needed. You should specify the timezone of the database, because that's what is used to determine which backups are to be copied to tape. Similarly, the keep_until_time should also specify the timezone of the database.
- If none of restore_until_scn, restore_until_time, or restore_point is specified, then archival backup is created by selecting the newest restorable backup within the past 14 days.
- If restore_tag is already used to create archival backup for the specified db_unique_name database, this API returns an error.

Syntax

```
PROCEDURE CREATE_ARCHIVAL_BACKUP(
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

```
db_unique_name IN VARCHAR2,
from_tag IN VARCHAR2 DEFAULT NULL,
compression_algorithm IN VARCHAR2 DEFAULT NULL,
encryption_algorithm IN VARCHAR2 DEFAULT NULL,
restore_point IN VARCHAR2 DEFAULT NULL,
restore_until_scn      IN VARCHAR2 DEFAULT NULL,
restore_until_time     IN TIMESTAMP WITH TIME ZONE DEFAULT NULL,
attribute_set_name     IN VARCHAR2,
format                 IN VARCHAR2 DEFAULT NULL,
autobackup_prefix      IN VARCHAR2 DEFAULT NULL,
restore_tag            IN VARCHAR2 DEFAULT NULL,
keep_until_time        IN TIMESTAMP WITH TIME ZONE DEFAULT NULL,
comments               IN VARCHAR2 DEFAULT NULL
max_redo_to_apply      IN NUMBER DEFAULT NULL);
```

Parameters

Table 25-9 CREATE_ARCHIVAL_BACKUP Parameters

Parameter	Description
db_unique_name	The DB_UNIQUE_NAME of the protected database. If this is not specified, or if the state of this database associated with this name is not valid, this API returns with an error.
from_tag	If specified, recovery appliance only considers backups using this tag for copying to tape. If invalid from_tag is specified, then API returns with an error.
compression_algorithm	If the backup is already compressed, this parameter is ignored, otherwise the output backup files will be compressed using specified algorithm. If invalid algorithm is specified, then API returns with an error. BASIC: Good compression ratios with potentially lower speed than MEDIUM. LOW: Optimized for speed with potentially lower compression raios than BASIC. MEDIUM: Recommended for most environments. Good combination of compression ratios and speed. HIGH: Best suited for operations over slower networks where the limiting factor is maximum network throughput. Provides the highest level of compression with the most negative impact on CPU performance. OFF: No compression. NULL: (default) existing value of compression algorithm is retained.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 25-9 (Cont.) CREATE_ARCHIVAL_BACKUP Parameters

Parameter	Description
encryption_algorithm	If the backup is already encrypted, this parameter is ignored. Otherwise the output backup files will be encrypted using specified algorithm. Valid values are 'AES128', 'AES192', 'AES256', 'OFF', 'CLIENT', or the constant equivalents ENC_OFF, ENC_AES128, ENC_AES192, ENC_AES256, ENC_CLIENT. **Note** A value of CLIENT or ENC_CLIENT requires the client to generate encrypted backups. Failure to do so will result in cloud backup job failures on the Recovery Appliance.
restore_point	User generated restore point name for which archival backups are created. If invalid restore point name is specified, this API returns an error..
restore_until_scn	User specified recovery SCN for which archival backups are created.
restore_until_time	User specified recovery time for which archival backups are created. Specify the timezone of the database.
attributre_set_name	User specified attribute set name. If an invalid attribute_set_name is specified, this returns an error.
format	The naming format of the output backup pieces. This parameter follows the same rules as the RMAN FORMAT parameter. If null, the default is defined by the queue_sbt_backup_task API.
autobackup_prefix	The original autobackup names will be given this prefix.
restore_tag	User specified tag for archival backups. If null, then tag values for archival backups will be the same as restore_point name.
keep_until_time	User specified retention time for the archival backup. If not specified, then the archival backup will be KEEP FOREVER backup. Specify the timezone of the database.
comments	Optional user supplied comment describing reason for executing this command.
max_redo_to_apply	User supplied parameter indicating the maximum number of days for which redo logs could be applied in order to create the archival backups. If a restore_until_time is specified, a full backup of the database must exist in the time range [restore_until_time - max_redo_to_apply, restore_until_time]. If restore_until_scn or restore_point is specified, or if no parameters are specified, then the current time is used and a full backup must exist in the time range [SYSTIMESTAMP - max_redo_to_apply, SYSTIMESTAMP]. If no value has been provided for the parameter then max_redo_to_apply defaults to 14 days.

Copyright © 2014, 2026, Oracle and/or its affiliates.

25.9 CREATE_POLLING_POLICY

This procedure creates a backup polling policy.

A backup polling policy specifies a directory where a protected database places incoming backups or archived redo log files. The policy also specifies the frequency with which the Recovery Appliance looks for backups in the polling location.

When the Recovery Appliance discovers a file through polling, the Recovery Appliance examines the file, and then uses its contents to associate it with a protected database that is registered with the Recovery Appliance. If the Recovery Appliance cannot associate the file with any registered protected database, then the Recovery Appliance logs a warning message and ceases to process the file.

Syntax

```
PROCEDURE create_polling_policy(
   polling_policy_name IN VARCHAR2,
   polling_location IN VARCHAR2,
   polling_frequency IN DSINTERVAL_UNCONSTRAINED DEFAULT NULL,
   delete_input IN BOOLEAN DEFAULT FALSE,
   comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-10 CREATE_POLLING_POLICY Parameters

Parameter	Description
`polling_policy_name`	The user-assigned name of the polling policy.
`polling_location`	The directory that the Recovery Appliance periodically examines for new backups. Do not specify this directory name in multiple polling policies.
`polling_frequency`	The frequency with which the Recovery Appliance examines the specified directory for new backups. System load may cause backup polling to occur less frequently. Specify the window as any valid `INTERVAL DAY TO SECOND` expression, such as `INTERVAL '2' DAY` (2 days), `INTERVAL '4' HOUR` (4 hours), and so on.
`delete_input`	The setting that controls deletion behavior. If `TRUE`, then the Recovery Appliance deletes files in the specified directory after copying them to a storage location. If `FALSE`, then the Recovery Appliance does not delete files that it discovers in the polling location.
`comments`	Optional user supplied comment describing reason for executing this command.

25.10 CREATE_PROTECTION_POLICY

This procedure creates a protection policy.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Syntax

```
PROCEDURE create_protection_policy (
   protection_policy_name IN VARCHAR2,
   description IN VARCHAR2 DEFAULT NULL,
   storage_location_name IN VARCHAR2,
   polling_policy_name IN VARCHAR2 DEFAULT NULL,
   recovery_window_goal IN DSINTERVAL_UNCONSTRAINED,
   max_retention_window IN DSINTERVAL_UNCONSTRAINED DEFAULT NULL,
   recovery_window_sbt IN DSINTERVAL_UNCONSTRAINED DEFAULT NULL,
   unprotected_window IN DSINTERVAL_UNCONSTRAINED DEFAULT NULL,
   guaranteed_copy IN VARCHAR2 DEFAULT 'NO',
   allow_backup_deletion IN VARCHAR2 DEFAULT 'YES',
   store_and_forward IN VARCHAR2 DEFAULT 'NO',
   log_compression_algorithm IN VARCHAR2 DEFAULT 'BASIC',
   autotune_reserved_space IN VARCHAR2 DEFAULT 'NO',
   recovery_window_compliance IN DSINTERVAL_UNCONSTRAINED DEFAULT NULL,
   keep_compliance IN VARCHAR2 'NO',
   comments IN VARCHAR2 DEFAULT NULL,
   max_reserved_space IN VARCHAR2 DEFAULT NULL,
   secure_mode IN VARCHAR2 DEFAULT 'NO',
   level0_refresh IN DSINTERVAL_UNCONSTRAINED DEFAULT NULL
);
```

Parameters

Table 25-11 CREATE_PROTECTION_POLICY Parameters

Parameter	Description
`protection_policy_name`	The user-assigned name of the protection policy.
`description`	An optional description of the usage for the policy.
`storage_location_name`	The name of the storage location. The Recovery Appliance uses this location for actively received incoming backups, and for newly created backup files for all databases sharing this protection policy.
`polling_policy_name`	The name of the backup polling policy. The polling policy specifies the rules for how the Recovery Appliance polls for backups of protected databases that use this protection policy. If null, then no backup polling occurs for databases that use this protection policy.
`recovery_window_goal`	The recovery window goal for databases that use this protection policy. For each protected database, the Recovery Appliance attempts to ensure that the oldest backup on disk can support a point-in-time recovery to any time within the specified interval, counting backward from the current time. Specify the goal as any valid `INTERVAL DAY TO SECOND` expression, such as `INTERVAL '2' DAY` (2 days), `INTERVAL '4' HOUR` (4 hours), and so on.
`max_retention_window`	The maximum length of time that the Recovery Appliance must retain backups for databases that use this protection policy. Recovery Appliance only holds backups longer than the specified period when they are required to preserve the recovery window goal for a database. If null, `max_retention_window` defaults to `2*recovery_window_goal`.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 25-11 (Cont.) CREATE_PROTECTION_POLICY Parameters

Parameter	Description
recovery_window_sbt	The recovery window for SBT backups of databases that use this protection policy. For each protected database, the Recovery Appliance keeps backups long enough on tape to guarantee that a recovery is possible to any time within the specified interval, counting backward from the current time. If this parameter is not null, then you must also create an SBT job for this protection policy, and then schedule it using a scheduling facility such as Oracle Scheduler. See CREATE_SBT_JOB_TEMPLATE. If this parameter is null, the purge backup automatically is never run and backups are kept beyond their expiration date. Specify the window as any valid INTERVAL DAY TO SECOND expression, such as INTERVAL '2' DAY (2 days), INTERVAL '4' HOUR (4 hours), and so on.
unprotected_window	The maximum amount of data loss that is tolerable for databases using this protection policy. When a protected database exceeds the specified amount of data loss, the Recovery Appliance posts a warning to RA_INCIDENT_LOG. The most recent time to which each protected database is recoverable is shown in the HIGH_TIME column of RA_RESTORE_RANGE. Specify the window as any valid INTERVAL DAY TO SECOND expression, such as INTERVAL '2' DAY (2 days), INTERVAL '4' HOUR (4 hours), and so on. The unprotected_window_threshold specifies the last set value for the unprotected_window in this API. unprotected_window specifies how much time is *"unprotected"* for this database. For this calculation, the data in ra_disk_restore_range does not have backup from online redo yet and is not reflected in ra_*restore_range views.
guaranteed_copy	The setting of the guaranteed copy feature. Specifying NO means that the Recovery Appliance always accepts new backups, even if it must delete old backups when space is low. This option prioritizes the ability to successfully process the backup currently being received over the ability to restore older backups. Specifying YES ensures that the Recovery Appliance copies backup data to tape or cloud before removing it from Recovery Appliance storage. This option prioritizes the ability to restore older backups over the ability to successfully process the backup currently being received. If set to YES, then for each protected database the Recovery Appliance can only hold up to disk_reserve_space bytes of backup data that is not yet copied to all libraries with the guaranteed_copy=YES. If hardware or network errors prevent timely copying, then future attempts to create new unbacked up pieces will fail when the Recovery Appliance reaches the disk_reserve_space limit.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 25-11 (Cont.) CREATE_PROTECTION_POLICY Parameters

Parameter	Description
allow_backup_deletion	Setting this to NO will prevent RMAN users from deleting backups on the Recovery Appliance. The default value is set to YES. NO means that the Recovery Appliance will prevent backups from being deleted by RMAN users for the databases using this protection policy. YES means that the Recovery Appliance will allow for backups to be deleted by RMAN users for the databases using this protection policy.
store_and_forward	The setting for the Backup and Redo Failover feature. This setting is used only in a protection policy defined on the alternate Recovery Appliance where the protected databases associated with this policy will redirect backups and redo in the event of an outage on the primary Recovery Appliance. YES means that the alternate Recovery Appliance does not index these redirected backups. Instead, the backups are stored as-is, and are sent to the primary Recovery Appliance when the outage is over. The backup pieces are deleted once they are replicated on the primary; support for incremental forever is turned off for this alternate Recovery Appliance only. The downstream Recovery Appliance resumes the incremental forever strategy once it receives these backups. NO is the default. Refer to Managing Temporary Outages with a Backup and Redo Failover Strategy for more information.
log_compression_algorithm	The setting for the archive log compression feature. This setting is used to adjust the compression level of NZDL/polled archive log backups. OFF means that the archive logs will not be compressed. BASIC means the BASIC compression algorithm will be used to compress the backups. LOW means the LOW compression algorithm will be used to compress the backups. MEDIUM means the MEDIUM compression algorithm will be used to compress the backups. HIGH means the HIGH compression algorithm will be used to compress the backups. Advanced Compression Option (ACO) license is not required on the protected database for use of LOW, MEDIUM, and HIGH log compression settings. For more details on log compression usage, see *"Changes in the Protection Policy Compression Algorithms"* (MOSnote Doc ID KB392785).

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 25-11 (Cont.) CREATE_PROTECTION_POLICY Parameters

Parameter	Description
autotune_reserved_space	This setting is used to control whether the Recovery Appliance will automatically define and update the reserved_space settings for databases associated with this policy. Even when this feature is enabled, initial and updated settings for reserved_space may still be supplied under the update_db to override the automatic modifications of reserved_space for a period specified by the recovery_window_goal parameter. YES means that the Recovery Appliance will supply an initial reserved_space setting for a database if none is supplied. The Recovery Appliance will also tune settings daily based upon database space usage. NO means that the Recovery Appliance administrator is responsible for specifying and maintaining the reserved_space settings for databases associated with this protection policy. NO is the default.j
recovery_window_compliance	This setting specifies for each database a range of backups that will not be deleted. These backups must not use more than disk_reserved_space bytes of storage, and if they do, new backups will be rejected until those backups age out of the range. "filling disk_reserved_space" should be "filling disk_reserved_space with compliance protected backups" Specify the window as any valid INTERVAL DAY TO SECOND expression, such as INTERVAL '4' HOUR (4 hours).
keep_compliance	This setting prevents someone from using RMAN CHANGE command to shrink the "keep until time" specified for an archival backup. YES means the "keep until time" for an archival backup may not be modified by the RMAN CHANGE command. If KEEP_COMPLIANCE is active, KEEP FOREVER backups will never be deleted. NO means the "keep until time" for an archival backup may be modified by the RMAN CHANGE command. NO is the default.
comments	Optional user supplied comment describing reason for executing this command.
max_reserved_space	This parameter is the maximum disk_reserved_space permitted for each database individually that is supported by the protection policy The format of this value is a character string that must contain a number consisting only of the characters 0-9, followed optionally by one of the following unit specifiers: • K: Kilobytes • M: Megabytes • G: Gigabytes • T: Terabytes If no unit is specified, then Recovery Appliance interprets the value as a number of bytes. If max_reserved_space is specified as NULL, the disk_reserved_space setting for the databases will not be contrained except by the restriction tha the sum of the reserved spaces for all databases must fit within the storage location.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 25-11 (Cont.) CREATE_PROTECTION_POLICY Parameters

Parameter	Description
secure_mode	Determines whether backups stored on the Recovery Appliance must be encrypted. YES means that only encrypted backup and redo are accepted by the Recovery Appliance. NO means unencrypted backups are allowed to be stored on the Recovery Appliance. NO is the default.
level0_refresh	If specified, the Recovery Appliance chooses some number of data files from each backup to be level 0 backups. This spreads the creation of new level 0 backup data across the level0_refresh interval. Specify the refresh cycle as any valid INTERVAL DAY TO SECOND expression, such as INTERVAL '20' DAY (20 days). If you set the value to 100 days, then 1% of the database will perform a level 0 backup each day. Effectively, once 100 days is complete, all datafiles will be level 0 with RA_FORMAT=TRUE (or *Block Pool Formatted*). The purpose of this option is to limit the number of Data Encryption Key (DEK) hashes required to restore a database. Each level 1 backup has a new DEK. During a restore, every block is sent to the client along with all the DEKs that are associated with them. During ingest, the DEKs are counted. When 65% of the maximum DEKs for the buffer is reached, a new level 0 refresh is set for that datafile.

25.11 CREATE_RESTORE_POINT

This procedure is used to define a named restore point on the Recovery Appliance that can then be used later to create an archival backup.

Syntax

```
PROCEDURE CREATE_ARCHIVAL_BACKUP(
   db_unique_name IN VARCHAR2,
   restore_point IN VARCHAR2,
   restore_until_scn      IN NUMBER DEFAULT NULL,
   restore_until_time     IN TIMESTAMP WITH TIME ZONE DEFAULT NULL,
   comments               IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-12 CREATE_ARCHIVAL_BACKUP Parameters

Parameter	Description
db_unique_name	The DB_UNIQUE_NAME of the protected database.
restore_point	User specified name for the restore point name.
restore_until_scn	User specified recovery SCN for which a restore point is created.
restore_until_time	User specified recovery time for which a restore point is created.
comments	Optional user supplied comment describing reason for executing this command.

Copyright © 2014, 2026, Oracle and/or its affiliates.

25.12 CREATE_REPLICATION_SERVER

This procedure defines a configuration for a downstream Recovery Appliance that forms part of a Recovery Appliance replication scheme.

This procedure creates metadata for the downstream Recovery Appliance, but does not replicate any backups. Use the ADD_REPLICATION_SERVER procedure to link the downstream Recovery Appliance to one or more protection policies, so that the Recovery Appliance sends backups for protected databases assigned to these policies to the downstream Recovery Appliance.

Syntax

```
PROCEDURE create_replication_server (
   replication_server_name IN VARCHAR2,
   sbt_so_name IN VARCHAR2,
   sbt_parms IN VARCHAR2 DEFAULT NULL,
   max_streams IN NUMBER DEFAULT NULL,
   catalog_user_name IN VARCHAR2,
   wallet_alias IN VARCHAR2,
   wallet_path IN VARCHAR2,
   proxy_url IN VARCHAR2 DEFAULT NULL,
   proxy_port IN NUMBER DEFAULT NULL,
   http_timeout IN NUMBER DEFAULT NULL,
   comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-13 CREATE_REPLICATION_SERVER Parameters

Parameter	Description
`replication_server_name`	The user-assigned name of the downstream Recovery Appliance. This value is converted to upper-case before storing.
`sbt_so_name`	The name and path to the Recovery Appliance Backup Module. The module is an Oracle-supplied media library that simulates an SBT device. The Recovery Appliance uses this library to communicate with the downstream Recovery Appliance.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 25-13 (Cont.) CREATE_REPLICATION_SERVER Parameters

Parameter	Description

`sbt_parms`	The name and path of a client configuration file in the form (`RA_<item>=`*file_system_location*). The parentheses are mandatory. The client configuration file is a text file.
	The following shows the sample contents of a client configuration file:
	`ra_host=oam2.example.com:6498`
	`ra_wallet='location=file:/u01/oracle/wallets credential_alias=repcred1'`
	Note
	The system parameters `SBT_LIBRARY`, `RA_WALLET`, `CREDENTIAL_ALIAS`, `PROXY_ULR`, and `PROXY_PORT` must be specified through the `create_replication_server` parameters and not through `sbt_parms`.

`max_streams`	The maximum number of simultaneous replication tasks. If null, which is the recommended setting, then the upstream Recovery Appliance determines the number of streams to use for replication based on the number of its nodes.
`catalog_user_name`	Ignored. Automatically populated with the Recovery Appliance Catalog Owner.
`wallet_alias`	The alias that identifies the credential within the wallet that the upstream Recovery Appliances uses to authenticate with the downstream Recovery Appliance.
`wallet_path`	The path to the local Oracle wallet (excluding the wallet file name). Path must start with `file:` .
`proxy_url`	The URL of any required proxy server, in the format *host*.
`proxy_port`	The port number of the proxy server.
`http_timeout`	The HTTP timeout interval, in seconds. Usually you leave this parameter set to null, to accept the system default HTTP timeout, unless directed to set it to a different value by Oracle Support.
`comments`	Optional user supplied comment describing reason for executing this command.

25.13 CREATE_SBT_ATTRIBUTE_SET

This procedure creates an SBT attribute set that SBT jobs can use.

An SBT attribute set provides a grouping of attributes that control the execution of an SBT job. These attributes enable you to specify settings for the media management library, including destination media pool or media family. You can define multiple SBT attribute sets. Multiple jobs can reference a single attribute set.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Syntax

```
PROCEDURE create_sbt_attribute_set(
   lib_name IN VARCHAR2,
   attribute_set_name IN VARCHAR2,
   streams IN NUMBER DEFAULT NULL,
   poolid IN NUMBER DEFAULT NULL,
   parms IN VARCHAR2 DEFAULT NULL,
   send IN VARCHAR2 DEFAULT NULL,
   comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-14 CREATE_SBT_ATTRIBUTE_SET Parameters

Parameter	Description
lib_name	The name of the SBT library to associate with the attribute set.
attribute_set_name	User-assigned name of the attribute set. Attribute set names must be unique.
streams	The maximum number of concurrent streams that the Recovery Appliance uses for automated backups. The number of concurrent streams never exceeds the limits set by the drives and restore_drives attributes of the SBT library. If streams is null, then the Recovery Appliance uses all available drives.
poolid	The media pool number to use as the destination for backup copies. This parameter accepts a value in the same format as the POOL parameter of the RMAN BACKUP command.
parms	The media management library-specific parameter string for the backup copy operation. The string has the same format as the PARMS option of the RMAN ALLOCATE CHANNEL command. During SBT backup operations for this attribute, the Recovery Appliance merges the value of this parameter with the PARMS parameter specified in the CREATE_SBT_LIBRARY procedure.
send	The string that the Recovery Appliance uses to send additional media management library-specific parameters for the backup copy operation. The string has the same format as the SEND option of the RMAN ALLOCATE CHANNEL command. During backup operations for this attribute, the Recovery Appliance merges the value of this parameter with the SEND parameter specified in the CREATE_SBT_LIBRARY procedure.
comments	Optional user supplied comment describing reason for executing this command.

25.14 CREATE_SBT_JOB_TEMPLATE

This procedure creates an SBT job that describes how the Recovery Appliance chooses backups for copying to tape/cloud. This form of this overloaded procedure applies to backups for all protected databases assigned to the specified protection policy.

After you create an SBT backup job, you must schedule it with a scheduling facility such as Oracle Scheduler. See QUEUE_SBT_BACKUP_TASK.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Syntax

```
PROCEDURE create_sbt_job_template (
   template_name            IN VARCHAR2,
   protection_policy_name   IN VARCHAR2,
   attribute_set_name       IN VARCHAR2,
   backup_type              IN VARCHAR2,
   full_template_name       IN VARCHAR2 DEFAULT NULL,
   from_tag                 IN VARCHAR2 DEFAULT NULL,
   priority                 IN NUMBER DEFAULT SBT_PRIORITY_MEDIUM,
   copies                   IN NUMBER DEFAULT 1,
   window                   IN DSINTERVAL_UNCONSTRAINED DEFAULT NULL,
   compression_algorithm    IN VARCHAR2 DEFAULT NULL,
   encryption_algorithm     IN VARCHAR2 DEFAULT NULL,
   comments                 IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-15 CREATE_SBT_JOB_TEMPLATE Parameters

Parameter	Description
`template_name`	The user-assigned name of this SBT job template.
`protection_policy_name`	The name of the protection policy to which this SBT job applies. Backups for all protected databases assigned to this protection policy are eligible for copying.
`attribute_set_name`	The name of the SBT attribute set to use for this SBT job.
`backup_type`	The types of backups that this SBT job chooses for copying to tape. The string must be a comma-separated list of the following types: `ALL`: Shorthand for `FULL`, `INCR`, `ARCH` `INCR`: Copies all incremental logs that have not yet been copied to tape, since the most recent full backup. `ARCH`: Copies all archived redo log backups that have not yet been copied to tape, since the most recent full backup. `FULL`: Copies the most recent virtual level 0 backup, if it has not already been copied, to tape. The backup can either be a virtual level 0 backup that is based on the most recent level 0 backup received or a virtual level 0 backup that is based on the most recent level 1 backup received, whichever is more recent.
`full_template_name`	The full name of this SBT job template. This applies only to `INCR` and `ARCH` backup types. The full name links full backups with the incremental backups and archived redo log files needed to recover them. If only one full backup template exists for the specified tape library, then this parameter defaults to the name of this template, which means it does not need to be specified. If multiple full backup templates exist, then you must specify the full template name. The specified `FULL` template name must belong to the same SBT library as the `INCR` or `ARCH` job. If `backup_type` is set to `FULL` or `ALL`, then `full_template_name` is the same as `template_name`.
`from_tag`	The tag name. If specified, then the Recovery Appliance only considers backups using this tag for copying to tape. Refer to *"Oracle Database Backup and Recovery Reference"* for the correct format of the TAG string.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 25-15 (Cont.) CREATE_SBT_JOB_TEMPLATE Parameters

Parameter	Description
priority	The priority of this job for tape resource usage. Lower priority values take precedence over higher values. 0 is the highest possible priority. You can use any number that is greater than or equal to 0. The pre-defined values are as follows: SBT_PRIORITY_LOW maps to 1000 SBT_PRIORITY_MEDIUM maps to 100 SBT_PRIORITY_HIGH maps to 10 SBT_PRIORITY_CRITICAL maps to 1 The default priority is SBT_PRIORITY_MEDIUM. Restore jobs by default have SBT_PRIORITY_CRITICAL priority.
copies	The number of distinct copies of each backup that this SBT job creates. Valid values range from 1 (default) to 4.
window	The window of time in which this job can copy backups to tape. Copy tasks that are not able to start within the specified window must wait until the next scheduled job execution.
compression_algorithm	Specifies the compression algorithm. If compression_algorithm is specified, it will override the compression algorithm defined in template_name for this single operation. If template_name is NULL, it defines the compression algorithm for this operation. BASIC: Good compression ratios with potentially lower speed than MEDIUM. LOW: Optimized for speed with potentially lower compression raios than BASIC. MEDIUM: Recommended for most environments. Good combination of compression ratios and speed. HIGH: Best suited for operations over slower networks where the limiting factor is maximum network throughput. Provides the highest level of compression with the most negative impact on CPU performance. OFF: No compression. NULL: (default) indicates that the algorithm defined in the SBT job template should be used.
encryption_algorithm	Encryption algorithm to use for SBT jobs. Valid values are 'AES128', 'AES192', 'AES256', 'OFF', 'CLIENT', or the constant equivalents ENC_OFF, ENC_AES128, ENC_AES192, ENC_AES256, ENC_CLIENT. **Note** A value of CLIENT or ENC_CLIENT requires the client to generate encrypted backups. Failure to do so will result in cloud backup job failures on the Recovery Appliance.
comments	Optional user supplied comment describing reason for executing this command.

Copyright © 2014, 2026, Oracle and/or its affiliates.

25.15 CREATE_SBT_JOB_TEMPLATE

This procedure creates a new SBT backup job. The job describes how the Recovery Appliance chooses backups for copying to tape/cloud. This form of this overloaded procedure applies to backups for a single protected database only, whereas the previous form applies to backups of all databases assigned to a specific protection policy. With the exception of this difference, this procedure and its parameters are identical to the alternative form of this procedure.

Syntax

```
PROCEDURE create_sbt_job_template (
   template_name IN VARCHAR2,
   db_unique_name IN VARCHAR2,
   attribute_set_name IN VARCHAR2,
   backup_type IN VARCHAR2,
   full_template_name IN VARCHAR2 DEFAULT NULL,
   from_tag IN VARCHAR2 DEFAULT NULL,
   priority IN NUMBER DEFAULT SBT_PRIORITY_MEDIUM,
   copies IN NUMBER DEFAULT 1,
   window IN DSINTERVAL_UNCONSTRAINED DEFAULT NULL,
   compression_algorithm IN VARCHAR2 DEFAULT NULL,
   encryption_algorithm IN VARCHAR2 DEFAULT NULL,
   comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-16 CREATE_SBT_JOB_TEMPLATE Parameters

Parameter	Description
db_unique_name	The DB_UNIQUE_NAME of the protected database to which this SBT job applies. This SBT job copies only backups that belong to the specified database.
comments	Optional user supplied comment describing reason for executing this command.

25.16 CREATE_SBT_LIBRARY

This procedure creates metadata describing an installed media management software library. The Recovery Appliance uses the specified library to copy backups from internal storage either to tape or to other tertiary storage supported by this media manager.

Syntax

```
PROCEDURE create_sbt_library (
   lib_name IN VARCHAR2,
   drives IN NUMBER,
   restore_drives IN NUMBER DEFAULT 0,
   parms IN VARCHAR2 DEFAULT NULL,
   send IN VARCHAR2 DEFAULT NULL,
   guaranteed IN VARCHAR2 DEFAULT 'NO',
   immutable IN VARCHAR2 DEFAULT 'NO',
   comments IN VARCHAR2 DEFAULT NULL,
   sbt_mirror IN VARCHAR@ DEFAULT 'NO');
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

Parameters

Table 25-17 CREATE_SBT_LIBRARY Parameters

Parameter	Description
lib_name	The user-specified name that the Recovery Appliance uses to refer to this SBT library.
drives	The maximum number of tape drives that this SBT library can access. The Recovery Appliance never uses more than the specified number of concurrent streams when accessing this library.
restore_drives	The number of tape drives that the Recovery Appliance reserves for restore operations. If specified, then the Recovery Appliance uses a maximum of drives - restore_drives drives for backup operations, which ensures that the Recovery Appliance always has the specified number of drives available for restore operations. If not specified, then the Recovery Appliance can use all available drives for backups, which means that a restore operation might have to wait for a drive to become free.
parms	The library-specific parameter string that the Recovery Appliance uses to access this SBT library. This string has the same format as the PARMS option of the RMAN ALLOCATE CHANNEL command. The string usually contains the SBT_LIBRARY parameter.
send	The parameter string that the Recovery Appliance uses to send additional library-specific parameters to this SBT library. This string has the same format as the SEND option of the RMAN ALLOCATE CHANNEL command.
guaranteed	If YES, this library may be used as a backing store to support the GUARANTEED_COPY protection policy attribute.
immutable	If YES, this library may be used as a backing store to support the KEEP_COMPLIANCE protection policy attribute.
comments	Optional user supplied comment describing reason for executing this command.
sbt_mirror	If YES, this library will be mapped to one or more remote SBT libraries on downstream Recovery Appliance using library name and type. Query the RA_SBT_MIRROR view for a list of mirrored SBT libraries.

> **Note**
>
> SBT mirroring is supported with bi-directional (or Backup Anywhere) replication mode. SBT mirroring is not supported in one-way replication mode. If the upstream Recovery Appliance has SBT mirroring and one-way replication, its backups are not accessible by the downstream Recovery Appliance.

25.17 DELETE_DB

This procedure deletes all local backups associated with this database from the Recovery Appliance. Backups on tape, in the cloud, or replicated are not affected.

Copyright © 2014, 2026, Oracle and/or its affiliates.

If the Recovery Appliance cannot delete the local backups owned by this database due to errors, then the DELETE_DB operation fails. If errors occur, then the specified database is not completely removed from the Recovery Appliance. The Recovery Appliance logs errors that occur during the DELETE_DB procedure in the RA_INCIDENT_LOG view. If the wait parameter is specified as TRUE, then the Recovery Appliance also raises these errors in the session in which the DELETE_DB is called. If you diagnose the errors and fix the problem, then you can run DELETE_DB again.

Syntax

```
PROCEDURE delete_db (
   db_unique_name       IN VARCHAR2,
   wait                 IN BOOLEAN DEFAULT TRUE,
   comments             IN VARCHAR2 DEFAULT NULL,
   delete_sbt           IN BOOLEAN DEFAULT FALSE);
```

Parameters

Table 25-18 DELETE_DB Parameters

Parameter	Description
db_unique_name	The DB_UNIQUE_NAME of the database to be removed.
wait	The wait behavior of the procedure. If TRUE, then the procedure will not return until the backups and metadata for the specified database are completely removed from the Recovery Appliance. If FALSE, then the procedure returns immediately, and the database deletion operation continues in the background.
comments	Optional user supplied comment describing reason for executing this command.
delete_sbt	If TRUE, the backups on tape and cloud will be deleted when the database is removed. If FALSE, the backups on tape and cloud will not be deleted when the database is removed. It is the responsibility of the DBA to clean up backups on tape and cloud at a later time.

25.18 DELETE_POLLING_POLICY

This procedure deletes the specified backup polling policy.

Syntax

```
PROCEDURE delete_polling_policy (
   polling_policy_name IN VARCHAR2,
   comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-19 DELETE_POLLING_POLICY Parameters

Parameter	Description
polling_policy_name	The name of the backup polling policy to delete.
comments	Optional user supplied comment describing reason for executing this command.

Copyright © 2014, 2026, Oracle and/or its affiliates.

25.19 DELETE_PROTECTION_POLICY

This procedure deletes the specified protection policy.

The specified policy must not be associated with any database.

Syntax

```
PROCEDURE delete_protection_policy (
   protection_policy_name IN VARCHAR2,
   comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-20 DELETE_PROTECTION_POLICY Parameters

Parameter	Description
`protection_policy_name`	The name of the protection policy to delete.
`comments`	Optional user supplied comment describing reason for executing this command.

25.20 DELETE_REPLICATION_SERVER

This procedure deletes a replication server configuration. The Recovery Appliance removes all metadata relating to the downstream Recovery Appliance.

Syntax

```
PROCEDURE delete_replication_server (
   replication_server_name IN VARCHAR2,
   force IN BOOLEAN DEFAULT FALSE,
   comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-21 DELETE_REPLICATION_SERVER Parameters

Parameter	Description
`replication_server_name`	The name of the replication server configuration to delete.
`force`	The deletion behavior when a protection policy is associated with the configuration. If FALSE, and if the replication server configuration is still associated with a protection policy, then the deletion fails. In this case, you must first call REMOVE_REPLICATION_SERVER. If TRUE, then `delete_replication_server` first removes the replication server configuration from the protection policy.
`comments`	Optional user supplied comment describing reason for executing this command.

Copyright © 2014, 2026, Oracle and/or its affiliates.

25.21 DELETE_SBT_ATTRIBUTE_SET

This procedure deletes the specified SBT attribute set.

Syntax

```
PROCEDURE delete_sbt_attribute_set(
   attribute_set_name IN VARCHAR2,
   comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-22 DELETE_SBT_ATTRIBUTE_SET Parameters

Parameter	Description
attribute_set_name	The name of the SBT attribute set to delete.
comments	Optional user supplied comment describing reason for executing this command.

25.22 DELETE_SBT_JOB_TEMPLATE

This procedure deletes the specified SBT job template.

Syntax

```
PROCEDURE delete_sbt_job_template (
   template_name IN VARCHAR2,
   comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-23 DELETE_SBT_JOB_TEMPLATE Parameters

Parameter	Description
template_name	The name of the SBT job to delete. The Recovery Appliance removes tasks belonging to this job from the task queue but does not terminate any executing task.
comments	Optional user supplied comment describing reason for executing this command.

25.23 DELETE_SBT_LIBRARY

This procedure deletes the metadata describing the specified SBT library.

The Recovery Appliance only removes the SBT library object, and does not uninstall the media management software.

This procedure deletes any SBT jobs and attributes created for this SBT library.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Syntax

```
PROCEDURE delete_sbt_library (
   lib_name IN VARCHAR2,
   comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-24 DELETE_SBT_LIBRARY Parameters

Parameter	Description
lib_name	The name of the SBT library to delete.
comments	Optional user supplied comment describing reason for executing this command.

25.24 ESTIMATE_SPACE

This procedure estimates the amount of storage in GB required for recovery of a given database and a desired recovery window. It requires a database name and a desired recovery window.

If the database is in an unregistered state or if there are no backups for the database, estimate_space returns NULL or no value.

Syntax

```
FUNCTION estimate_space (
   db_unique_name IN VARCHAR2,
   target_window IN DSINTERVAL_UNCONSTRAINED,
   comments IN VARCHAR2 DEFAULT NULL)
RETURN NUMBER;
```

Parameters

Table 25-25 ESTIMATE_SPACE Parameters

Parameter	Description
db_unique_name	The name of the database needing the storage estimate.
target_window	The desired recovery window for the database. Specify the goal as any valid INTERVAL DAY TO SECOND expression, such as INTERVAL '2' DAY (2 days), INTERVAL '4' HOUR (4 hours), and so on.
comments	Optional user supplied comment describing reason for executing this command.
RETURNS	Gigabytes of space needed to achieve recoverability across the target_window.

Copyright © 2014, 2026, Oracle and/or its affiliates.

25.25 GET_REDO_TRANSPORT_LAG

Returns the current redo transport lag on the specified database.

Syntax

```
FUNCTION get_redo_transport_lag (
   db_unique_name      IN VARCHAR2,
   comments            IN VARCHAR2 DEFAULT NULL)
RETURN DSINTERVAL_UNCONSTRAINED;
```

Parameters

Table 25-26 GET_REDO_TRANSPORT_LAG Parameters

Parameter	Description
db_unique_name	The name of the database experiencing the transport lag.
comments	Optional user supplied comment describing reason for executing this command.
RETURNS	DSINTERVAL_UNCONSTRAINED.

25.26 GRANT_DB_ACCESS

This procedure grants the necessary privileges to the specified recovery Appliance user account to enable this account to back up, restore, and access recovery catalog metadata for the specified protected database.

Syntax

```
PROCEDURE grant_db_access (
   username           IN VARCHAR2,
   db_unique_name     IN VARCHAR2,
   comments           IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-27 GRANT_DB_ACCESS Parameters

Parameter	Description
username	The name of the Recovery Appliance user account.
db_unique_name	The protected database for which the privilege is being granted.
comments	Optional user supplied comment describing reason for executing this command.

25.27 KEY_REKEY

This procedure rekeys encryption keys for all databases with existing encryption keys.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Syntax

```
PROCEDURE key_rekey(,
   comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-28 KEY_REKEY Parameters

Parameter	Description
comments	Optional user supplied comment describing reason for executing this command.

25.28 KEY_REKEY

This procedure rekeys encryption keys for the specified database with an existing encryption key.

Syntax

```
PROCEDURE key_rekey (
   db_unique_name       IN VARCHAR2,
   comments             IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-29 KEY_REKEY Parameters

Parameter	Description
db_unique_name	The DB_UNIQUE_NAME of the database to generate a new encryption key. Note: this routine will not create a new key, only rekey an existing key
comments	Optional user supplied comment describing reason for executing this command.

25.29 KEY_REKEY

This procedure rekeys encryption keys for all databases with existing encryption keys in the specified protection_policy

Syntax

```
PROCEDURE key_rekey (
   protection_policy_name IN VARCHAR2,
   comments IN VARCHAR2 DEFAULT NULL);
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

Parameters

Table 25-30 KEY_REKEY Parameters

Parameter	Description
protection_policy_name	Generate new encryption keys for databases that are part of this protection policy.
comments	Optional user supplied comment describing reason for executing this command.

25.30 MIGRATE_TAPE_BACKUP

This procedure makes pre-migration tape backups accessible to the Recovery Appliance through the specified SBT library. You must first import metadata about the tape backups into the Recovery Appliance catalog using the RMAN IMPORT CATALOG command.

This procedure performs the metadata adjustments required to access pre-existing tape backups, but does not physically move backups. The pre-existing backups must already be accessible by the specified SBT library.

Syntax

```
PROCEDURE migrate_tape_backup(
   db_unique_name       IN VARCHAR2,
   sbt_lib_name         IN VARCHAR2,
   comments             IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-31 MIGRATE_TAPE_BACKUP Parameters

Parameter	Description
db_unique_name	The comma-delimited list of protected databases whose backups are to be migrated. You must already have registered each db_unique_name with the Recovery Appliance catalog, and have added it to the Recovery Appliance with ADD_DB.
sbt_lib_name	The SBT library that the Recovery Appliance uses to access existing tape backups for the specified protected database. See CREATE_SBT_LIBRARY.
comments	Optional user supplied comment describing reason for executing this command.

25.31 MOVE_BACKUP

This procedure moves one or more long-term archival backup pieces from the Recovery Appliance to an SBT destination.

The Recovery Appliance copies all backup pieces matching the specified tag to the location specified with the format and template_name parameters. After the Recovery Appliance copies each backup piece successfully, the Recovery Appliance deletes the backup piece from its original location.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Syntax

```
PROCEDURE move_backup (
   tag IN VARCHAR2,
   format IN VARCHAR2,
   template_name IN VARCHAR2,
   compression_algorithm IN VARCHAR2 DEFAULT NULL,
   encryption_algorithm IN VARCHAR2 DEFAULT NULL,
   comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-32 MOVE_BACKUP Parameters

Parameter	Description
`tag`	The tag of backups to copy. The Recovery Appliance removes all backups matching this tag.
`format`	The naming format of the backup pieces to create. This parameter follows the same rules as the RMAN `FORMAT` parameter. If not specified, default format is `RA_SBT_%d_%I_<SBT_job_template_key>_%U_<bs_key>`.
`template_name`	The name of the SBT job library template. The Recovery Appliance copies the backup piece to tape (or cloud), using the media pool referenced in the SBT template name as the copy destination.
`compression_algorithm`	Specifies the compression algorithm If `compression_algorithm` is specified, it will override the compression algorithm defined in `template_name` for this single operation. If `template_name` is NULL, it defines the compression algorithm for this operation. `BASIC`: Good compression ratios with potentially lower speed than `MEDIUM`. `LOW`: Optimized for speed with potentially lower compression raios than `BASIC`. `MEDIUM`: Recommended for most environments. Good combination of compression ratios and speed. `HIGH`: Best suited for operations over slower networks where the limiting factor is maximum network throughput. Provides the highest level of compression with the most negative impact on CPU performance. `OFF`: No compression. `NULL`: (default) indicates that the algorithm defined in the SBT job template should be used.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 25-32 (Cont.) MOVE_BACKUP Parameters

Parameter	Description
`encryption_algorithm`	Specifies the encryption algorithm If `encryption_algorithm` is specified, it will override the encryption algorithm defined in `template_name` for this single operation. If `template_name` is NULL, it defines the encryption algorithm for this operation. Valid values are 'AES128', 'AES192', 'AES256', 'OFF', 'CLIENT', or the constant equivalents ENC_OFF, ENC_AES128, ENC_AES192, ENC_AES256, ENC_CLIENT. **Note** A value of `CLIENT` or `ENC_CLIENT` requires the client to generate encrypted backups. Failure to do so will result in cloud backup job failures on the Recovery Appliance.
`comments`	Optional user supplied comment describing reason for executing this command.

25.32 MOVE_BACKUP_PIECE

This procedure moves a single long-term archival backup piece from the Recovery Appliance to an SBT destination.

The Recovery Appliance copies the specified backup piece to the location specified with the `format` and `template_name` parameters. After the Recovery Appliance copies the backup piece successfully, the Recovery Appliance deletes the backup piece from its original location.

Syntax

```
PROCEDURE move_backup_piece (
   bp_key IN NUMBER,
   format IN VARCHAR2,
   template_name IN VARCHAR2,
   compression_algorithm IN VARCHAR2 DEFAULT NULL,
   encryption_algorithm IN VARCHAR2 DEFAULT NULL,
   comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-33 MOVE_BACKUP_PIECE Parameters

Parameter	Description
`bp_key`	The unique key of the backup piece to move. Obtain this key from the `RC_BACKUP_PIECE` view.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 25-33 (Cont.) MOVE_BACKUP_PIECE Parameters

Parameter	Description
`format`	The naming format of the backup pieces to create. This parameter follows the same rules as the RMAN `FORMAT` parameter. If not specified, default format is `RA_SBT_%d_%I_<SBT_job_template_key>_%U_<bs_key>`.
`template_name`	The name of the SBT job library template. The Recovery Appliance copies the backup piece to tape, using the media pool referenced in the SBT template name as the copy destination.
`compression_algorithm`	Specifies the compression algorithm. If `compression_algorithm` is specified, it will override the compression algorithm defined in `template_name` for this single operation. If `template_name` is NULL, it defines the compression algorithm for this operation. `BASIC`: Good compression ratios with potentially lower speed than `MEDIUM`. `LOW`: Optimized for speed with potentially lower compression raios than `BASIC`. `MEDIUM`: Recommended for most environments. Good combination of compression ratios and speed. `HIGH`: Best suited for operations over slower networks where the limiting factor is maximum network throughput. Provides the highest level of compression with the most negative impact on CPU performance. `OFF`: No compression. `NULL`: (default) indicates that the algorithm defined in the SBT job template should be used.
`encryption_algorithm`	Specifies the encryption algorithm If `encryption_algorithm` is specified, it will override the encryption algorithm defined in `template_name` for this single operation. If `template_name` is NULL, it defines the encryption algorithm for this operation. Valid values are 'AES128', 'AES192', 'AES256', 'OFF', 'CLIENT', or the constant equivalents ENC_OFF, ENC_AES128, ENC_AES192, ENC_AES256, ENC_CLIENT. **Note** A value of `CLIENT` or `ENC_CLIENT` requires the client to generate encrypted backups. Failure to do so will result in cloud backup job failures on the Recovery Appliance.
`comments`	Optional user supplied comment describing reason for executing this command.

Copyright © 2014, 2026, Oracle and/or its affiliates.

25.33 PAUSE_REPLICATION_DATABASE

This procedure pauses replication for the specified database with all associated replication servers. If `replication_server_name` is specified, replication for the one database/one replication server is paused.

The Recovery Appliance permits in-progress replication of backup pieces to complete. If the Recovery Appliance queued backup pieces for replication through this replication server configuration but did not replicate them, then the Recovery Appliance holds the backup pieces until you call RESUME_REPLICATION_DATABASE. No replication tasks that run against this database/Recovery Appliance can execute until you `resume_replication_database` for the specified database/downstream Recovery Appliance.

Syntax

```
PROCEDURE pause_replication_database (
   db_unique_name           IN VARCHAR2,
   replication_server_name  IN VARCHAR2 DEFAULT NULL,
   comments                 IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-34 PAUSE_REPLICATION_DATABASE Parameters

Parameter	Description
`db_unique_name`	The protected database for which to pause replication.
`replication_server_name`	If not null, replication is paused only for the one database on this specified replication server. If null, replication is paused for that database on all associated replication servers.
`comments`	Optional user supplied comment describing reason for executing this command.

25.34 PAUSE_REPLICATION_SERVER

This procedure pauses replication to the specified downstream Recovery Appliance.

The Recovery Appliance permits in-progress replication of backup pieces to complete. If the Recovery Appliance queued backup pieces for replication through this replication server configuration but did not replicate them, then the Recovery Appliance holds the backup pieces until you call RESUME_REPLICATION_SERVER. No replication tasks that run against this Recovery Appliance can execute until you resume replication to the downstream Recovery Appliance.

Syntax

```
PROCEDURE pause_replication_server  (
   replication_server_name IN VARCHAR2,
   comments IN VARCHAR2 DEFAULT NULL);
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

Parameters

Table 25-35 PAUSE_REPLICATION_SERVER Parameters

Parameter	Description
`replication_server_name`	The name of the downstream Recovery Appliance.
`comments`	Optional user supplied comment describing reason for executing this command.

25.35 PAUSE_SBT_LIBRARY

This procedure pauses the specified SBT library. The Recovery Appliance allows in-progress copies of backup pieces to complete. However, if backup pieces were queued for copy through this SBT library but not yet copied, then the Recovery Appliance holds them until you resume the SBT library. No new SBT jobs that run against this library can execute until you resume the library (RESUME_SBT_LIBRARY).

Query the `RA_SBT_LIBRARY` view for a list of existing SBT libraries.

Syntax

```
PROCEDURE pause_sbt_library(
   lib_name IN VARCHAR2,
   comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-36 PAUSE_SBT_LIBRARY Parameters

Parameter	Description
`lib_name`	Name of the SBT library to pause.
`comments`	Optional user supplied comment describing reason for executing this command.

25.36 POPULATE_BACKUP_PIECE

This procedure pushes the specified backup piece into the delta store.

Use this procedure to initially populate the delta store or to correct corruption in the delta store. The delta store is backup data that supports an incremental-forever backup solution. Only incremental backups (not `KEEP` backups) can become part of the delta store.

Syntax

```
PROCEDURE populate_backup_piece(
   backup_piece_key IN NUMBER,
   comments IN VARCHAR2 DEFAULT NULL);
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

Parameters

Table 25-37 POPULATE_BACKUP_PIECE Parameters

Parameter	Description
backup_piece_key	Either the backup piece key provided by the Recovery Appliance when it detected a corruption, or the backup piece to insert into the delta store. If the key represents a virtual backup piece, then the Recovery Appliance searches for a backup piece to resolve corruption in the delta store. If the key does not represent a virtual backup, then the Recovery Appliance inserts this backup piece into the delta store. This backup must be an incremental backup that is not a KEEP backup.
comments	Optional user supplied comment describing reason for executing this command.

25.37 QUEUE_SBT_BACKUP_TASK

This procedure queues the backup pieces selected by the specified SBT job template for copying to tape. Typically, a scheduling utility such as Oracle Scheduler calls this procedure.

Syntax

```
PROCEDURE queue_sbt_backup_task(
   template_name IN VARCHAR2,
   format IN VARCHAR2 DEFAULT NULL,
   autobackup_prefix IN VARCHAR2 DEFAULT NULL,
   tag IN VARCHAR2 DEFAULT NULL,
   comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-38 QUEUE_SBT_BACKUP_TASK Parameters

Parameter	Description
template_name	The name of the SBT job template that specifies the backup pieces to copy to tape.
format	The naming format of the backup pieces to create. This parameter follows the same rules as the RMAN FORMAT parameter. If not specified, default format is RA_SBT_%d_%I_<SBT_job_template_key>_%U_<bs_key>.
autobackup_prefix	The original autobackup names will be prefixed with this autobackup_prefix.
tag	User specified tag for backups to be copied See CREATE_SBT_JOB_TEMPLATE.
comments	Optional user supplied comment describing reason for executing this command.

Copyright © 2014, 2026, Oracle and/or its affiliates.

25.38 REMOVE_REPLICATION_SERVER

This procedure removes the specified replication server configuration from the specified protection policy. After the operation succeeds, the Recovery Appliance no longer replicates backups of databases protected by this policy to the downstream Recovery Appliance.

Syntax

```
PROCEDURE remove_replication_server (
   replication_server_name IN VARCHAR2,
   protection_policy_name IN VARCHAR2,
   comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-39 REMOVE_REPLICATION_SERVER Parameters

Parameter	Description
`replication_server_name`	The name of the replication server configuration to remove.
`protection_policy_name`	The name of the protection policy from which the specified replication server configuration is to be removed.
`comments`	Optional user supplied comment describing reason for executing this command.

25.39 RENAME_DB

This procedure changes the name of the specified protected database in the Recovery Appliance metadata.

Use this procedure when the `DB_UNIQUE_NAME` for a protected database changes, so that the Recovery Appliance metadata reflects the correct name.

Syntax

```
PROCEDURE rename_db (
   db_unique_name_old  IN VARCHAR2,
   db_unique_name_new  IN VARCHAR2,
   comments            IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-40 RENAME_DB Parameters

Parameter	Description
`db_unique_name_old`	The `DB_UNIQUE_NAME` to change.
`db_unique_name_new`	The new `DB_UNIQUE_NAME`.
`comments`	Optional user supplied comment describing reason for executing this command.

Copyright © 2014, 2026, Oracle and/or its affiliates.

25.40 RESET_ERROR

This procedure modifies the specified set of incident log entries to have the status RESET. It takes multiple optional input parameters to allow bulk resetting of errors. When two or more input parameters are specified in a single RESET_ERROR call, only records that match all the input parameters specified together are RESET. Errors marked in this fashion do not cause Oracle Enterprise Manager to raise alerts. If the Recovery Appliance determines that the problem is still occurring, then errors that have been reset change to ACTIVE status. The primary use of this API is to reset the error status for nonrecurring errors, such as transient media failures.

Syntax

```
PROCEDURE reset_error(
   incident# NUMBER DEFAULT NULL,
   error_code NUMBER DEFAULT NULL,
   error_text VARCHAR2 DEFAULT NULL,
   task_id NUMBER DEFAULT NULL,
   component VARCHAR2 DEFAULT NULL,
   low_time TIMESTAMP WITH TIME ZONE DEFAULT NULL,
   high_time TIMESTAMP WITH TIME ZONE DEFAULT NULL,
   comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-41 RESET_ERROR Parameters

Parameter	Description
incident#	The unique identifier of the incident log entry to reset.
error_code	The error code of the incident log entry to reset.
error_text	The text of the error message that completely or partially matches the incident log entry to reset.
task_id	The identifier for the task of the incident log entry to reset.
component	The component of the incident log entry to reset.
low_time	
high_time	The low and high time with respect to last_seen column of the incident log entry to reset. If only one of them or neither of them are provided then the default values are used. The default value for low_time is the year 2000. The default value for high_time is systimestamp. Obtain the identifiers from the RA_INCIDENT_LOG view.
comments	Optional user supplied comment describing reason for executing this command.

25.41 RESUME_DB

This procedure restores a suspended database to normal operation. Only suspended databases may be resumed.

Suspended databases must be resumed before they can be backed up. A new reserved space value is required if the database is suspended in order to state how much space the newly

Copyright © 2014, 2026, Oracle and/or its affiliates.

reinstated database now requires. The reserved space value is not required if the protection policy for the database has `autotune_reserved_space='YES'`.

When renaming a databse in a replication configuration, make sure to call this command on both Recovery Appliances involved with replication.

Syntax

```
PROCEDURE resume_db (
   db_unique_name IN VARCHAR2,
   reserved_space IN VARCHAR2 DEFAULT NULL,
   comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-42 RESUME_DB Parameters

Parameter	Description
`db_unique_name`	The `DB_UNIQUE_NAME` of the database to be resumed.
`reserved_space`	Refer to ADD_DB
`comments`	Optional user supplied comment describing reason for executing this command.

25.42 RESUME_REPLICATION_DATABASE

This procedure resumes replication for the specified database after a previous call to `pause_replication_database`.

Syntax

```
PROCEDURE resume_replication_database (
   db_unique_name IN VARCHAR2,
   replication_server_name IN VARCHAR2 DEFAULT NULL,
   comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-43 RESUME_REPLICATION_DATABASE Parameters

Parameter	Description
`db_unique_name`	The protected database for which to pause replication.
`replication_server_name`	If not null, replication is resumed only for the one database on this specified replication server. If null, replication is resumed for that database on all associated replication servers.
`comments`	Optional user supplied comment describing reason for executing this command.

25.43 RESUME_REPLICATION_SERVER

This procedure resumes replication to the specified downstream Recovery Appliance, after a previous call to PAUSE_REPLICATION_SERVER.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Syntax

```
PROCEDURE resume_replication_server (
   replication_server_name IN VARCHAR2,
   comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-44 RESUME_REPLICATION_SERVER Parameters

Parameter	Description
replication_server_name	The name of the downstream Recovery Appliance.
comments	Optional user supplied comment describing reason for executing this command.

25.44 RESUME_SBT_LIBRARY

This procedure resumes a paused SBT library.

Query the RA_SBT_LIBRARY to determine which SBT libraries are paused (see PAUSE_SBT_LIBRARY).

Syntax

```
PROCEDURE resume_sbt_library(
   lib_name IN VARCHAR2,
   comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-45 RESUME_SBT_LIBRARY Parameters

Parameter	Description
lib_name	Name of the SBT library to resume.
comments	Optional user supplied comment describing reason for executing this command.

25.45 REVOKE_DB_ACCESS

This procedure revokes privileges on one protected database from the specified Recovery Appliance user account.

Syntax

```
PROCEDURE revoke_db_access (
   username            IN VARCHAR2,
   db_unique_name      IN VARCHAR2,
   comments            IN VARCHAR2 DEFAULT NULL);
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

Parameters

Table 25-46 REVOKE_DB_ACCESS Parameters

Parameter	Description
username	The name of the user account from which to revoke the privilege.
db_unique_name	The protected database for which the privilege is being revoked.
comments	Optional user supplied comment describing reason for executing this command.

25.46 SET_SYSTEM_DESCRIPTION

This procedure sets a descriptive name for users to apply to their Recovery Appliance. The name provided here will be seen in the RA_SERVER view.

Syntax

```
PROCEDURE set_system_description(
   sys_desc VARCHAR2,
   comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-47 SET_SYSTEM_DESCRIPTION Parameters

Parameter	Description
sys_desc	A descriptive name for this Recovery Applicance.
comments	Optional user supplied comment describing reason for executing this command.

25.47 SHUTDOWN

Synonymous with SHUTDOWN_RECOVERY_APPLIANCE.

Syntax

```
PROCEDURE shutdown(
  timeout IN DSINTERVAL_UNCONSTRAINED DEFAULT NULL,
  comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-48 SHUTDOWN Parameters

Parameter	Description
timeout	Maximum time interval allowed for the orderly shutdown before the operations becomes an ABORT_RECOVERY_APPLIANCE

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 25-48 (Cont.) SHUTDOWN Parameters

Parameter	Description
comments	Optional user supplied comment describing reason for executing this command.

25.48 SHUTDOWN_RECOVERY_APPLIANCE

This procedure performs a clean shutdown of the Recovery Appliance.

This procedure permits in-progress operations to complete before shutting down. The shutdown can take some time to complete. The wait for operations to complete can be constrained by specifying the timeout parameter.

If an immediate shutdown is required, then use ABORT_RECOVERY_APPLIANCE.

Syntax

```
PROCEDURE shutdown_recovery_appliance(
  timeout IN DSINTERVAL_UNCONSTRAINED DEFAULT NULL,
  comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-49 SHUTDOWN_RECOVERY_APPLIANCE Parameters

Parameter	Description
timeout	Maximum time interval allowed for the orderly shutdown before the operations becomes an ABORT_RECOVERY_APPLIANCE
comments	Optional user supplied comment describing reason for executing this command.

25.49 STARTUP

Synonymous with STARTUP_RECOVERY_APPLIANCE.

Syntax

```
PROCEDURE startup(
  comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-50 STARTUP Parameters

Parameter	Description
comments	Optional user supplied comment describing reason for executing this command.

Copyright © 2014, 2026, Oracle and/or its affiliates.

25.50 STARTUP_RECOVERY_APPLIANCE

This procedure starts the Recovery Appliance after it has been shut down or terminated.

The Recovery Appliance can process backup and restore requests only when it is started.

If the Recovery Appliance was started with STARTUP_RECOVERY_APPLIANCE, and if any instance of the Recovery Appliance metadata database is restarted, then a database startup trigger automatically restarts the Recovery Appliance. The only exception is when the metadata database is restarted with the `RESETLOGS` option, which requires you to run the `startup_recovery_appliance` procedure to repair corrupt metadata.

Syntax

```
PROCEDURE startup_recovery_appliance(
  comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-51 STARTUP_RECOVERY_APPLIANCE Parameters

Parameter	Description
`comments`	Optional user supplied comment describing reason for executing this command.

25.51 SUSPEND_DB

This procedure deletes all local disk backups associated with this database from the Recovery Appliance. Backups on tape, in the cloud, or replicated to other Recovery Appliances are not affected.

While a database is suspended, it will not accept backups. The database must be resumed before it can return to normal operation.

A suspended database does not have a `reserved_space`.

If the Recovery Appliance cannot delete the local backups owned by this database due to errors, then the `SUSPEND_DB` operation fails. If errors occur, then the backups from the specified database are not completely removed from the Recovery Appliance local storage. The Recovery Appliance logs errors that occur during the `SUSPEND_DB` procedure in the `RA_INCIDENT_LOG` view. If the `wait` parameter is specified as `TRUE`, then the Recovery Appliance also raises these errors in the session in which the `SUSPEND_DB` is called. If you diagnose the errors and fix the problem, then you can run `SUSPEND_DB` again.

Syntax

```
PROCEDURE suspend_db (
   db_unique_name       IN VARCHAR2,
   wait                 IN BOOLEAN DEFAULT TRUE,
   comments             IN VARCHAR2 DEFAULT NULL);
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

Parameters

Table 25-52 SUSPEND_DB Parameters

Parameter	Description
db_unique_name	The DB_UNIQUE_NAME of the database to be suspended.
wait	The wait behavior of the procedure. If TRUE, then the procedure will not return until the backups and metadata for the specified database are completely removed from the Recovery Appliance. If FALSE, then the procedure returns immediately, and the database deletion operation continues in the background.
comments	Optional user supplied comment describing reason for executing this command.

25.52 UPDATE_ARCHIVAL_BACKUP_KEEP

This procedure updates the retention time of archival backup with the specified keep_until_time. Archival backup is identified by user specified restore_tag and restore_point.

This API has the following restrictions for input options:

- If restore point doesn't exist for the specified restore_tag and restore_point of the db_unique_name database, this returns an error.
- If keep_compliance of the protection policy is set to YES and keep_until_time is less than the existing retention time of archival backup, this returns an error.

Syntax

```
PROCEDURE update_archival_backup_keep(
   db_unique_name       IN VARCHAR2,
   restore_tag          IN VARCHAR2,
   restore_point        IN VARCHAR2 DEFAULT NULL,
   restore_until_scn    IN VARCHAR2,
   keep_until_time      IN TIMESTAMP WITH TIME ZONE,
   comments             IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-53 UPDATE_ARCHIVAL_BACKUP_KEEP Parameters

Parameter	Description
db_unique_name	The DB_UNIQUE_NAME of the protected database. If this is not specified, or if the state of this database is not valid, this API returns with an error.
restore_tag	Specifies the user-defined tag of the archival backup.
restore_point	User specified restore point name of the archival backup.
keep_until_time	User specified retention time for the archival backup. If specified as NULL, then the archival backup will be KEEP FOREVER backup.
comments	Optional user supplied comment describing reason for executing this command.

Copyright © 2014, 2026, Oracle and/or its affiliates.

25.53 UPDATE_DB

This procedure changes the attributes that are assigned to the specified protected database.

Syntax

```
PROCEDURE update_db (
   db_unique_name           IN VARCHAR2,
   protection_policy_name   IN VARCHAR2 DEFAULT NULL,
   reserved_space           IN VARCHAR2 DEFAULT dbms_ra_misc.varchar2null('p2'),
   db_timezone              IN VARCHAR2 DEFAULT NULL,
   incarnations             IN VARCHAR2 DEFAULT 'CURRENT',
   skip_initial_replication IN BOOLEAN DEFAULT FALSE,
   compliance_hold          IN TIMESTAMP WITH TIME ZONE DEFAULT
dbms_ra_misc.tsnull('pl'),
   comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-54 UPDATE_DB Parameters

Parameter	Description
db_unique_name	The DB_UNIQUE_NAME of the database.
protection_policy_name	The name of the protection policy to assign to the database. The protection policy must exist. The new protection policy controls new storage operations. If the old and new protection policies specify different storage locations, the Recovery Appliance starts a background task to move data from the old storage location to the new location. Recovery Appliance only moves backups that are not obsolete. If a move between storage locations is required, then the RA_DATABASE view does not show the new protection policy until the move has started. If the Recovery Appliance must perform higher priority work, then the Recovery Appliance may not start the move for several hours.
reserved_space	See ADD_DB.
db_timezone	The time zone where this database is located. By default, protected databases are assigned to the same time zone as the Recovery Appliance. If the protected database is in a different time zone, then use this procedure to assign the database to the correct time zone.
incarnations	Comma-delimited list of keys for all previous database incarnations to update the db_timezone. By default, this procedure updates the current incarnation. If the protected database administrator has not specified a db_timezone and this list contains the current incarnation key, then this procedure associates the time zone with the metadata for the current incarnation. If the time zone is set in the parameter file already for the incarnation, then the new db_timezone is ignored.
skip_initial_replication	If set to TRUE, the initial replication is skipped.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 25-54 (Cont.) UPDATE_DB Parameters

Parameter	Description
compliance_hold	The time from which backups may not be deleted from the Recovery Appliance or guaranteed tape or cloud storage. The database must be recoverable to the time specified by this compliance_hold. Specify the time as any valid TIMESTAMP WITH TIME ZONE expression, such as SYSTIMESTAMP - NUMTODSINTERVAL(7, 'DAY'), meaning *"starting 7 days ago."* If the database is being updated with compliance_hold, make sure that its associated protection policies do **not** have autotune_reserved_space configured.
comments	Optional user supplied comment describing reason for executing this command.

25.54 UPDATE_POLLING_POLICY

This procedure modifies the parameters for an existing backup polling policy.

Parameters that are NULL retain their existing values.

Syntax

```
PROCEDURE update_polling_policy (
   polling_policy_name IN VARCHAR2,
   polling_location IN VARCHAR2 DEFAULT NULL,
   polling_frequency IN DSINTERVAL_UNCONSTRAINED DEFAULT NULL,
   delete_input IN BOOLEAN DEFAULT NULL,
   comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-55 UPDATE_POLLING_POLICY Parameters

Parameter	Description
polling_policy_name	The name of the backup polling policy to update.
polling_location	See CREATE_POLLING_POLICY.
polling_frequency	See CREATE_POLLING_POLICY.
delete_input	See CREATE_POLLING_POLICY.
comments	Optional user supplied comment describing reason for executing this command.

25.55 UPDATE_PROTECTION_POLICY

This procedure modifies the parameters for an existing protection policy.

If a parameter is NULL, its value remains unchanged, except as noted below.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Syntax

```
PROCEDURE update_protection_policy (
   protection_policy_name IN VARCHAR2,
   description IN VARCHAR2 DEFAULT NULL,
   storage_location_name IN VARCHAR2 DEFAULT NULL,
   polling_policy_name IN VARCHAR2 DEFAULT dbms_ra_misc.varchar2null('p1'),
   recovery_window_goal IN DSINTERVAL_UNCONSTRAINED DEFAULT NULL,
   max_retention_window IN DSINTERVAL_UNCONSTRAINED DEFAULT
dbms_ra_misc.intervalnull('p3'),
   recovery_window_sbt IN DSINTERVAL_UNCONSTRAINED DEFAULT
dbms_ra_misc.intervalnull('p2'),
   unprotected_window IN DSINTERVAL_UNCONSTRAINED DEFAULT
dbms_ra_misc.intervalnull('p4'),
   guaranteed_copy IN VARCHAR2 DEFAULT NULL,
   allow_backup_deletion IN VARCHAR2 DEFAULT NULL,
   store_and_forward IN VARCHAR2 DEFAULT NULL,
   log_compression_algorithm IN VARCHAR2 DEFAULT NULL,
   autotune_reserved_space IN VARCHAR2 DEFAULT 'NO',
   recovery_window_compliance IN DSINTERVAL_UNCONSTRAINED DEFAULT
dbms_ra_misc.intervalnull('p5'),
   keep_compliance IN VARCHAR2 'NO',
   comments IN VARCHAR2 DEFAULT NULL,
   max_reserved_space IN VARCHAR2 DEFAULT dbms_ra_misc.varchar2nul('p6'),
   secure_mode IN VARCHAR2 DEFAULT 'NULL',
   level0_refresh IN DSINTERVAL_UNCONSTRAINED DEFAULT
dbms_ra_misc.varchar2nul('p6')
);
```

Parameters

Table 25-56 UPDATE_PROTECTION_POLICY Parameters

Parameter	Description
protection_policy_name	The name of the protection policy to update.
description	See CREATE_PROTECTION_POLICY.
storage_location_name	See CREATE_PROTECTION_POLICY. If you change the storage location for this protection policy, then the Recovery Appliance starts background jobs to move data from the old storage location to the new storage location.
polling_policy_name	See CREATE_PROTECTION_POLICY. If you do not specify this parameter, then the policy retains the existing value. If you specify a value (including null), then the Recovery Appliance sets the new value.
recovery_window_goal	See CREATE_PROTECTION_POLICY.
max_retention_window	See CREATE_PROTECTION_POLICY. If this parameter is not specified, its old value is retained. If specified, including being specified as NULL, the new value is set.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 25-56 (Cont.) UPDATE_PROTECTION_POLICY Parameters

Parameter	Description
recovery_window_sbt	See CREATE_PROTECTION_POLICY. If you do not specify this parameter, then the policy retains the existing value. If you specify a value (including null), then the Recovery Appliance sets the new value.
unprotected_window	If you do not specify this parameter, then the policy retains the existing value. If you specify a value (including null), then the Recovery Appliance sets the new value. See CREATE_PROTECTION_POLICY.
guaranteed_copy	See CREATE_PROTECTION_POLICY.
allow_backup_deletion	See CREATE_PROTECTION_POLICY.
store_and_forward	See CREATE_PROTECTION_POLICY.
log_compression_algorithm	See CREATE_PROTECTION_POLICY.
autotune_reserved_space	See CREATE_PROTECTION_POLICY.
recovery_window_compliance	See CREATE_PROTECTION_POLICY. This setting specifies for each database a range of backups that will not be deleted. These backups must not use more than disk_reserved_space bytes of storage, and if they do, new backups will be rejected until those backups age out of the range. Specify the window as any valid INTERVAL DAY TO SECOND expression, such as INTERVAL '4' HOUR (4 hours).
keep_compliance	See CREATE_PROTECTION_POLICY. YES means the "keep until time" for an archival backup may not be modified by the RMAN CHANGE command. NO means the "keep until time" for an archival backup may be modified by the RMAN CHANGE command. NO is the default.
comments	Optional user supplied comment describing reason for executing this command.
max_reserved_space	This parameter is the maximum disk_reserved_space permitted for each database individually that is supported by the protection policy The format of this value is a character string that must contain a number consisting only of the characters 0-9, followed optionally by one of the following unit specifiers: • K: Kilobytes • M: Megabytes • G: Gigabytes • T: Terabytes • P: Petabytes If no unit is specified, then Recovery Appliance interprets the value as a number of bytes. If max_reserved_space is specified as NULL, the disk_reserved_space setting for databases will not be constrained except by the restriction that the sum of the reserved spaces for all databases must fit within the storage location.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 25-56 (Cont.) UPDATE_PROTECTION_POLICY Parameters

Parameter	Description
secure_mode	Determines whether backups stored on the Recovery Appliance must be encrypted. YES means that only encrypted backup and redo are accepted by the Recovery Appliance. NO means unencrypted backups are allowed to be stored on the Recovery Appliance. NO is the default.
level0_refresh	If specified, the Recovery Appliance chooses some number of data files from each backup to be level 0 backups. This spreads the creation of new level 0 backup data across the level0_refresh interval. Its purpose is to limit the number of encryption keys needed to maintain virtual level 0 backups. Specify the refresh cycle as any valid INTERVAL DAY TO SECOND expression, such as INTERVAL '20' DAY (20 days).

25.56 UPDATE_REPLICATION_SERVER

This procedure changes the settings for a replication server configuration.

Note the following restrictions for changing replication server parameters:

The configuration does not retain the sbt_parms string from the original CREATE_REPLICATION_SERVER call. If you change any parameter other than max_streams, then you must pass in this value.

Changing any setting other than max_streams requires replication to be paused, which you can achieve by calling PAUSE_REPLICATION_SERVER.

Parameters other than sbt_parms whose values are null remain unchanged, except as noted in the following parameter descriptions.

Syntax

```
PROCEDURE update_replication_server (
   replication_server_name IN VARCHAR2,
   sbt_so_name IN VARCHAR2 DEFAULT NULL,
   sbt_parms IN VARCHAR2 DEFAULT NULL,
   max_streams IN NUMBER DEFAULT dbms_ra_misc.number2null('p4'),
   catalog_user_name IN VARCHAR2 DEFAULT NULL,
   wallet_alias IN VARCHAR2 DEFAULT NULL,
   wallet_path IN VARCHAR2 DEFAULT dbms_ra_misc.varchar2null('p1'),
   proxy_url IN VARCHAR2 DEFAULT dbms_ra_misc.varchar2null('p2'),
   proxy_port IN NUMBER DEFAULT dbms_ra_misc.number2null('p3'),
   http_timeout IN NUMBER DEFAULT NULL,
   comments IN VARCHAR2 DEFAULT NULL);
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

Parameters

Table 25-57 UPDATE_REPLICATION_SERVER Parameters

Parameter	Description
`replication_server_name`	The name of the replication server configuration to update. This value is converted to upper-case before storing.
`sbt_so_name`	See CREATE_REPLICATION_SERVER.
`sbt_parms`	See CREATE_REPLICATION_SERVER.
`max_streams`	See CREATE_REPLICATION_SERVER. If you do not specify this parameter, then the Recovery Appliance retains the existing value. If you specify a value (including null), then the Recovery Appliance sets the new value.
`catalog_user_name`	See CREATE_REPLICATION_SERVER.
`wallet_alias`	See CREATE_REPLICATION_SERVER.
`wallet_path`	See CREATE_REPLICATION_SERVER. If you do not specify this parameter, then the Recovery Appliance retains the existing value. If you specify a value (including null), then the Recovery Appliance sets the new value. Path must start with `file:` .
`proxy_url`	See CREATE_REPLICATION_SERVER. If you do not specify this parameter, then the Recovery Appliance retains the existing value. If you specify a value (including null), then the Recovery Appliance sets the new value.
`proxy_port`	See CREATE_REPLICATION_SERVER. If you do not specify this parameter, then the Recovery Appliance retains the existing value. If you specify a value (including null), then the Recovery Appliance sets the new value.
`http_timeout`	See CREATE_REPLICATION_SERVER.
`comments`	Optional user supplied comment describing reason for executing this command.

25.57 UPDATE_SBT_ATTRIBUTE_SET

This procedure updates the parameters for the specified SBT attribute set.

If a parameter is null, then its value remains unchanged, except as noted in the following parameter descriptions.

Syntax

```
PROCEDURE update_sbt_attribute_set(
   attribute_set_name IN VARCHAR2,
   streams IN NUMBER DEFAULT dbms_ra_misc.number2null('p1'),
   poolid IN NUMBER DEFAULT NULL,
   parms IN VARCHAR2 DEFAULT dbms_ra_misc.varchar2null('p2'),
   send IN VARCHAR2 DEFAULT dbms_ra_misc.varchar2null('p3'),
   comments IN VARCHAR2 DEFAULT NULL);
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

Parameters

Table 25-58 UPDATE_SBT_ATTRIBUTE_SET Parameters

Parameter	Description
attribute_set_name	The name of the SBT attribute set to update.
streams	See CREATE_SBT_ATTRIBUTE_SET. If you do not specify this parameter, then the Recovery Appliance retains the existing value. If you specify a value (including null), then the Recovery Appliance sets the new value.
poolid	See CREATE_SBT_ATTRIBUTE_SET.
parms	See CREATE_SBT_ATTRIBUTE_SET. If you do not specify this parameter, then the Recovery Appliance retains the existing value. If you specify a value (including null), then the Recovery Appliance sets the new value.
send	See CREATE_SBT_ATTRIBUTE_SET. If you do not specify this parameter, then the Recovery Appliance retains the existing value. If you specify a value (including null), then the Recovery Appliance sets the new value.
comments	Optional user supplied comment describing reason for executing this command.

25.58 UPDATE_SBT_JOB_TEMPLATE

This procedure updates the parameters for the specified SBT job.

If a parameter is null, then its value remains unchanged, except as noted in the following parameter descriptions.

Syntax

```
PROCEDURE update_sbt_job_template (
   template_name IN VARCHAR2,
   attribute_set_name IN VARCHAR2 DEFAULT NULL,
   backup_type IN VARCHAR2 DEFAULT NULL,
   from_tag IN VARCHAR2 DEFAULT dbms_ra_misc.varchar2null('p1'),
   priority IN NUMBER DEFAULT NULL,
   copies IN NUMBER DEFAULT NULL,
   window IN DSINTERVAL_UNCONSTRAINED DEFAULT dbms_ra_misc.intervalnull('p2'),
   compression_algorithm IN VARCHAR2 DEFAULT NULL,
   encryption_algorithm IN VARCHAR2 DEFAULT NULL,
   comments IN VARCHAR2 DEFAULT NULL);
```

Parameters

Table 25-59 UPDATE_SBT_JOB_TEMPLATE Parameters

Parameter	Description
template_name	The name of the SBT job template to update.
attribute_set_name	See CREATE_SBT_JOB_TEMPLATE.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 25-59 (Cont.) UPDATE_SBT_JOB_TEMPLATE Parameters

Parameter	Description
backup_type	See CREATE_SBT_JOB_TEMPLATE.
from_tag	See CREATE_SBT_JOB_TEMPLATE. If you do not specify this parameter, then the Recovery Appliance retains the existing value. If you specify a value (including null), then the Recovery Appliance sets the new value.
priority	See CREATE_SBT_JOB_TEMPLATE.
copies	See CREATE_SBT_JOB_TEMPLATE.
window	See CREATE_SBT_JOB_TEMPLATE. If you do not specify this parameter, then the Recovery Appliance retains the existing value. If you specify a value (including null), then the Recovery Appliance sets the new value.
compression_algorithm	see CREATE_SBT_JOB_TEMPLATE. If a value is specified, it becomes the new setting. Specify OFF to remove the compression algorithm from this template. This changes the value of COMPRESSION_ALGORITHM to NONE. If you do not specify this parameter or if you specify NULL, then the Recovery Appliance retains the existing value.
encryption_algorithm	see CREATE_SBT_JOB_TEMPLATE.
comments	Optional user supplied comment describing reason for executing this command.

25.59 UPDATE_SBT_LIBRARY

This procedure modifies the parameters for the specified SBT library.

If a parameter is null, then its value remains unchanged, except as noted in the parms and send descriptions.

Syntax

```
PROCEDURE update_sbt_library (
   lib_name IN VARCHAR2,
   drives IN NUMBER DEFAULT NULL,
   restore_drives IN NUMBER DEFAULT NULL,
   parms IN VARCHAR2 DEFAULT dbms_ra_misc.varchar2null('p1'),
   send IN VARCHAR2 DEFAULT dbms_ra_misc.varchar2null('p2'),
   guaranteed IN VARCHAR2 DEFAULT 'NO',
   immutable IN VARCHAR2 DEFAULT 'NO',
   comments IN VARCHAR2 DEFAULT NULL,
   sbt_mirror IN VARCHAR2 DEFAULT NULL);
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

Parameters

Table 25-60 UPDATE_SBT_LIBRARY Parameters

Parameter	Description
lib_name	The name of the SBT library whose parameters are to be modified.
drives	See CREATE_SBT_LIBRARY.
restore_drives	See CREATE_SBT_LIBRARY.
parms	See CREATE_SBT_LIBRARY. If you do not specify this parameter, then the Recovery Appliance retains the existing value. If you specify a value (including null), then the Recovery Appliance sets the new value.
send	See CREATE_SBT_LIBRARY. If you do not specify this parameter, then the Recovery Appliance retains the existing value. If you specify a value (including null), then the Recovery Appliance sets the new value.
guaranteed	If YES, this library may be used as a backing store to support the GUARANTEED_COPY protection policy attribute.
immutable	If YES, this library may be used as a backing store to support the KEEP_COMPLIANCE protection policy attribute.
comments	Optional user supplied comment describing reason for executing this command.
sbt_mirror	If YES, this library will be mapped to one or more remote SBT libraries on downstream Recovery Appliance using library name and type. Query the RA_SBT_MIRROR view for a list of mirrored SBT libraries.

> **Note**
>
> SBT mirroring is supported with bi-directional (or Backup Anywhere) replication mode. SBT mirroring is not supported in one-way replication mode. If the upstream Recovery Appliance has SBT mirroring and one-way replication, its backups are not accessible by the downstream Recovery Appliance.

25.60 UPDATE_STORAGE_LOCATION

This procedure allocates additional space for the specified storage location. You cannot reduce the amount of space used by a storage location.

Syntax

```
PROCEDURE update_storage_location (
   storage_location_name IN VARCHAR2,
   comments IN VARCHAR2 DEFAULT NULL,
   autotune_space_limit IN VARCHAR2 DEFAULT dbms_ra_misc.varchar2null('pl'));
```


Copyright © 2014, 2026, Oracle and/or its affiliates.

Parameters

Table 25-61 UPDATE_STORAGE_LOCATION Parameters

Parameter	Description
`storage_location_name`	The name of the storage location to update.
`comments`	Optional user supplied comment describing reason for executing this command.
`autotune_space_limit`	Autotune will not limit reserved space growth when the total reserved space usage is below this limit. When the total reserved space usage is above this limit, autotune will limit subsequent reserved space growth to 10% per week for each database in the storage location. The format of this value is a character string that must contain a number consisting only of the characters `0-9`, followed optionally by one of the following unit specifiers: • `K`: Kilobytes • `M`: Megabytes • `G`: Gigabytes • `T`: Terabytes If no unit is specified, then Recovery Appliance interprets the value as a number of bytes. This value may be set to `NULL` if there should never be restrictions on reserved space growth through the `autotune_reserved_space` option.

Copyright © 2014, 2026, Oracle and/or its affiliates.

26

Recovery Appliance View Reference

Table 26-1 Recovery Appliance Views

Recovery Appliance View	Description
RA_ACTIVE_SESSION	This view lists information about active client sessions currently running in the Recovery Appliance.
RA_API_HISTORY	This view describes the history of user-issued API commands.
RA_CONFIG	This view lists the user configuration settings.
RA_DATABASE	This view lists the databases protected by this Recovery Appliance.
RA_DATABASE_HISTORY	This view has one row for each database for each day going back one year. There may be rows missing in case the DB_STATS_REFRESH task was delayed in its execution for over a day. There may also be phantom rows for deleted databases.
RA_DATABASE_STORAGE_USAGE	This view lists the storage usage for each protected database.
RA_DATABASE_SYNONYM	This view lists the protected databases and their equivalent names.
RA_DB_ACCESS	This view describes which Recovery Appliance user accounts have access to which protected databases.
RA_DISK_RESTORE_RANGE	The restore range of each protected database from disk backups on this Recovery Appliance.
RA_EM_SBT_JOB_TEMPLATE	This view lists defined SBT jobs and their statuses for Oracle Enterprise Manager.
RA_ENCRYPTION_INFO	This view describes the historical encryption key information.
RA_INCIDENT_LOG	This view describes the Recovery Appliance incidents.
RA_INCOMING_BACKUP_PIECES	This view describes the backup pieces being received by the Recovery Appliance.
RA_POLLING_FILES	This view lists the files backed up by Recovery Appliance from the polling location.
RA_POLLING_POLICY	This view lists the defined backup polling policies.
RA_PROTECTION_POLICY	This view lists the protection policies defined for this Recovery Appliance.
RA_PURGING_QUEUE	This view describes the order in which protected databases will have their oldest backups deleted when space is low.
RA_RECOVERY_COMPLIANCE	This view lists information about active compliance windows on the database in which backups may not be deleted.
RA_REPLICATION_CONFIG	This view lists the replication server configurations.
RA_REPLICATION_DATABASE	This view lists information on replication servers and protected databases.
RA_REPLICATION_PAIR	This view lists replication information for replicating protection policies.
RA_REPLICATION_POLICY	This view lists the association of replication servers to protection policy.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 26-1 (Cont.) Recovery Appliance Views

Recovery Appliance View	Description
RA_REQUEST BACKUP	This view describes the list of requested backup pieces from a remote Recovery Appliance.
RA_RESTORE_RANGE	This view describes the restore range of each protected database from all backups on this Recovery Appliance.
RA_SBT_ATTRIBUTE_SET	This view describes the defined SBT attribute set.
RA_SBT_JOB	This view describes the defined SBT job templates.
RA_SBT_LIBRARY	This view lists the defined SBT libraries.
RA_SBT_MIRROR	This view lists the SBT libraries that are being mirrored with remote Recovery Appliances.
RA_SBT_RESTORE_RANGE	This view describes the restore range of each database from SBT backups on the Recovery Appliance.
RA_SBT_TASK	This view lists the queued background SBT tasks and their run statuses.
RA_SBT_TEMPLATE_MDF	This view lists missing level 0 data file backups for each SBT template.
RA_SERVER	This view describes the current settings for the Recovery Appliance.
RA_STORAGE_HISTOGRAM	This view describes the storage allocation history for recent time periods.
RA_STORAGE_LOCATION	This view lists defined Recovery Appliance storage locations and their allocations.
RA_STORAGE_LOCATION_HISTORY	This view contains a daily history of storage location for the for the previous year. It has one row for each storage location for each day going back a year. It may contain phantom rows for storage locations that were deleted.
RA_TASK	This view lists queued background tasks and their run statuses.
RA_TIMER_TASK	This view describes timer process tasks and their planned executions.
RA_TIME_USAGE	This view describes the Recovery Appliance elapsed and idle time for the last 30 days.

26.1 RA_ACTIVE_SESSION

This view lists information about active client sessions currently running in the Recovery Appliance.

Column	Data Type	NULL	Description
INST_ID	NUMBER		The Recovery Appliance instance number where this session is running.
INSTANCE_NAME	VARCHAR2(16)		The Recovery Appliance instance name where this session is running.
HOST_NAME	VARCHAR2(64)		The Recovery Appliance host name where this session is running.
SID	NUMBER		The session ID for the active session.
SERIAL#	NUMBER		The session serial number, which uniquely identifies the objects in a session.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Column	Data Type	NULL	Description
SPID	VARCHAR2(24)		The operating system process identifier.
DB_KEY	NUMBER		The primary key for this database in the recovery catalog.
DB_UNIQUE_NAME	VARCHAR2(30)		The unique database name.
SBT_SID	VARCHAR2(64)		The SBT session identifier.
CLIENT_IDENTIFIER	VARCHAR2(64)		The client Identifier of the session.
MODULE	VARCHAR2(64)		The name of the module that is currently executing.
ACTION	VARCHAR2(64)		The name of the action that is currently executing.
SQL_ID	VARCHAR2(13)		The SQL identifier of the SQL statement that is currently being executed.
EVENT	VARCHAR2(64)		The resource or event for which the session is waiting.
P1	NUMBER		First wait event parameter
P2	NUMBER		The second wait event parameter.
P3	NUMBER		The third wait event parameter.
WAIT_TIME	NUMBER		The wait time in hundredths of a second. See description of V$SESSION.WAIT_TIME for more information.
SECONDS_IN_WAIT	NUMBER		The wait time (in seconds). If the session is currently waiting, then the value is the amount of time waited for the current wait. If the session is not in a wait, then the value is the amount of time since the start of the most recent wait.
STATE	VARCHAR2(19)		The state of the wait event: WAITING, WAITED UNKNOWN TIME, WAITED SHORT TIME, WAITED KNOWN TIME. See description of V$SESSION.STATE for more information.
TASK_ID	NUMBER		The task identifier.
TASK_TYPE	VARCHAR2(30)		The task type.
PRIORITY	NUMBER		The task priority.
TASK_STATE	VARCHAR2(13)		The processing state for the task: EXECUTABLE, RUNNING, COMPLETED, TASK_WAIT, FAILED, and so on.
JOB_NAME	VARCHAR2(128)		The DBMS_SCHEDULER job name.
TENANT_NAME	VARCHAR2		The tenant name for this database in the Recovery Appliance.
TENANT_IDENTIFIER	NUMBER		The customer tenant identifier for this database in the Recovery Appliance.

26.2 RA_API_HISTORY

This view describes the history of user-issued API commands.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Column	Data Type	NULL	Description
RESULTS	VARCHAR2(1000)		The results from running this command: SUCCESS or FAIL.
EXECUTE_TIME	TIMESTAMP(6) WITH TIME ZONE		The time at which the command started.
TASK_NAME	VARCHAR2(30)		The name of the task.
COMMAND_ISSUED	VARCHAR2(4000)		The full command as submitted by the user.
ELAPSED_SECONDS	NUMBER		The elapsed run time (in seconds) for the task.
API_USER	VARCHAR2(1000)		The user who initiated the API .

26.3 RA_CONFIG

This view lists the Recovery Appliance configuration settings. The list includes public configuration settings as well as internal configuration settings that have been altered from the default value.

In software release 21.1, changes made to the config table are tracked, as well as default values, which are the "best values" that the Recovery Appliance is shipped with. The view shows all non-underscore config entries as well as those that have values different from the default values.

Column	Data Type	NULL	Description
NAME	VARCHAR2(30)	NOT NULL	The name of the configuration variable. See "RA_CONFIG" for variable definitions and default values.
VALUE	VARCHAR2(100)		The value of the configuration variable.
DEFAULT_VALUE	VARCHAR2(100)		The default value of the configuration variable.
CHANGE_DATE	TIMESTAMP(6) WITH TIME ZONE		The date this configuration variable was changed from the default value.
COMMENTS	VARCHAR2(300)		The comment given at the time the configuration variable was changed.

26.4 RA_DATABASE

This view lists the databases protected by this Recovery Appliance.

Column	Data Type	NULL	Description
DB_UNIQUE_NAME	VARCHAR2(32)		The unique name of this protected database.
DB_KEY	NUMBER		The primary key for this database in the Recovery Appliance metadata database.
DELETING	VARCHAR2(7)		YES if this database is currently being deleted.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Column	Data Type	NULL	Description
STATE	VARCHAR2(12)		ACTIVE if this database is currently in active use. DELETING if this database is currently being deleted. SUSPENDING if this database is currently being suspended SUSPENDED if this database is currently suspended. UNREGISTERED if this database is currently unregistered. PROVISIONAL if a database is created under a AUTOTUNE_RESERVED_SPACE policy that cannot allocate its initial reserved space because there is no room in the storage locaiton.
DBID	NUMBER		The DBID for this protected database.
TENANT_NAME	VARCHAR2		The tenant name for this database in the Recovery Appliance.
TENANT_CSI	NUMBER		The customer tenant identifier for this database in the Recovery Appliance.
CREATION_TIME	TIMESTAMP(6) WITH TIME ZONE		The time when this database was added to the Recovery Appliance.
POLICY_NAME	VARCHAR2(128)		The name of the protection policy used by this database.
STORAGE_LOCATION	VARCHAR2(128)		The name of the Recovery Appliance storage location used by this protected database.
RECOVERY_WINDOW_GOAL	INTERVAL DAY(9) TO SECOND(6)		The recovery window goal for backups on disk, as specified in the protection policy.
MAX_RETENTION_WINDOW	INTERVAL DAY(9) TO SECOND(6)		The maximum amount of time to retain disk backups. The Recovery Appliance deletes disk backups when they are older than this window. However, backups may be retained longer if deleting them would negatively affect the recovery_window_goal requirement.
RECOVERY_WINDOW_SBT	INTERVAL DAY(9) TO SECOND(6)		The recovery window for backups on tape, as specified in the protection policy.
TIMEZONE	VARCHAR2(64)		The time zone offset of the protected database.
SPACE_USAGE	NUMBER		The amount of disk space (in GB) currently used by this protected database.
KEEP_SPACE	NUMBER		The amount of KEEP backup space (in GB) currently used by this database.
DISK_RESERVED_SPACE	NUMBER		The amount of disk space (in GB) reserved for the exclusive use of this database

Copyright © 2014, 2026, Oracle and/or its affiliates.

Column	Data Type	NULL	Description
AUTOTUNE_DISK_RESERVED_SPACE	VARCHAR2(3)		YES: The Recovery Appliance will automatically set and update DISK_RESERVED_SPACE as needed. NO: The administrator of the Recovery Appliance must set and update the DISK_RESERVED_SPACE manually. INTERRUPTED: Note: the DISK_RESERVED_SPACE has been manually altered. This state remains until RECOVERY_WINDOW_GOAL days after the last time DISK_RESERVED_SPACE was explicitly set. During the INTERRUPTED state, autotune will never modify the reserved space to a value below what it was previously set at.
DISK_RESERVED_SPACE_SET	TIMESTAMP(6) WITH TIME ZONE		The last time that DISK_RESERVED_SPACE was updated by the administrator of the Recovery Appliance.
GUARANTEED_COPY	VARCHAR2(3)		The status of the guaranteed copy setting: YES means that the Recovery Appliance replicates backups or copies them to tape before deleting them; NO means that the Recovery Appliance accepts new backups even if old backups must be purged because free space is low.
CUMULATIVE_USAGE	NUMBER		The cumulative amount of disk space (in GB) allocated for all backups received for this database.
REPLICATION_USAGE	NUMBER		The cumulative amount of disk space (in GB) replicated for this protected database.
CLOUD_USAGE	NUMBER		The cumulative amount of disk space (in GB) sent to cloud storage for this protected database.
SBT_USAGE	NUMBER		The cumulative amount of disk space (in GB) sent to SBT from this protected database.
REPLICATION_SETUP_STATUS	VARCHAR2(7)		The status of the setup for the downstream replication appliance for this database.
LAST_OPTIMIZE	TIMESTAMP(6) WITH TIME ZONE		The time when the most recent data placement optimization was completed.
LAST_VALIDATE	TIMESTAMP(6) WITH TIME ZONE		The time when the most recent validation of backup data was completed.
LAST_METADATA_VALIDATE	TIMESTAMP(6) WITH TIME ZONE		The time when the most recent validation of metadata was completed.
LAST_CROSSCHECK	TIMESTAMP(6) WITH TIME ZONE		The time when the most recent crosscheck of backup data was completed.
STORAGE_LOCATION_COUNT	NUMBER		The number of storage locations used by this database. If greater than one, then a storage location movement operation is in progress for this database.
STORAGE_MOVEMENT_PHASE	VARCHAR2(18)		The phase of the storage location movement operation for this protected database.
SIZE_ESTIMATE	NUMBER		The estimated size of the entire protected database (in GB). This does not refer to the space used by this database on the Recovery Appliance

Copyright © 2014, 2026, Oracle and/or its affiliates.

Column	Data Type	NULL	Description
RECOVERY_WINDOW_SPACE	NUMBER		The estimated space (in GB) that is needed to meet the recovery window goal.
RESTORE_WINDOW	INTERVAL DAY(9) TO SECOND(9)		The time range of backups used to compute the value of RECOVERY_WINDOW_SPACE.
DEDUPLICATION_FACTOR	NUMBER		The ratio of the total size of virtual full backups to the actual consumed space on the appliance for this protected database.
MINIMUM_RECOVERY_NEEDED	INTERVAL DAY(9) TO SECOND(9)		The minimum interval needed to restore any part of the protected database to the present if there are sufficient archive logs to perform the recovery.
UNPROTECTED_WINDOW_THRESHOLD	INTERVAL DAY(9) TO SECOND(6)		The user-specified maximum amount of data loss for protected databases that are subject to a protection policy. The Recovery Appliance generates an alert if the unprotected window of this database exceeds this value.
UNPROTECTED_WINDOW	INTERVAL DAY(9) TO SECOND(9)		The point beyond which recovery is impossible unless additional redo is available.
NZDL_ACTIVE	VARCHAR2(3)		YES if real-time redo transport is active. NO if redo has not recently been received.
ALLOW_BACKUP_DELETION	VARCHAR2(3)		The setting that controls whether RMAN backups for databases that use this protection policy can be deleted: NO means that the Recovery Appliance does not allow deletion of these backups; YES means that the Recovery Appliance allows deletion of these backups. Note that this parameter is available only with Zero Data Loss Recovery Appliance software update 12.1.1.1.7 and later.
STORE_AND_FORWARD	VARCHAR2(3)		The status of the Backup and Redo Failover setting: YES means that the Recovery Appliance applies the Backup and Redo Failover strategy to backups from the databases associated with this protection policy; NO means that the Backup and Redo Failover feature is not enabled and the Recovery Appliance applies the normal incremental-forever backup strategy instead. Note that this setting is available only with Zero Data Loss Recovery Appliance software update 12.1.1.1.8 and later.
AUTOTUNE_RESERVED_SPACE	VARCHAR2(3)		YES: Recovery Appliance automatically sets and updates disk_reserved_space as needed. NO: The administrator of the Recovery Appliance must set and update disk_reserved_space manually.
ENCRYPTION_TAG	VARCHAR2(128)		Identifying tag for this database's encryption key. Defaults to $<db_unique_name>_$<dbid>
ENCRYPTION_KEYID	VARCHAR2(78)		The keyid for this encryption key. It may be used for lookup in the key vault.
COMPLIANCE_HOLD	TIMESTAMP(6) WITH TIME ZONE		Backups that are created after this time stamp may not be deleted.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Column	Data Type	NULL	Description
RECOVERY_WINDOW_COMPLIANCE	INTERVAL DAY(9) TO SECOND(6)		Time interval during which backups may not be deleted after they are created.
KEEP_COMPLIANCE	VARCHAR2(3)		YES: The Recovery Appliance will prevent the deletion of KEEP backups. NO: The administrator of the Recovery Appliance is permitted to remove KEEP backups.
LAST_PURGE	TIMESTAMP(6) WITH TIME ZONE		Time of last purge attempt for database.
COPYALL_STATE			NONE This database is not receiving replication COPYALL backups. IN PROGRESS This database is in the process of receiving replicated COPYALL backups. COPIED This database has received all replicated COPYALL backups and is indexing them. COMPLETE This database has received and processed all replication COPYALL backups.
COPYALL_SOURCE			The name of the upstream Recovery Applicance replicating COPYALL backups.
SECURE_MODE			YES Backups must be encrypted. NO Backups may be stored unencrypted.
LEVEL0_REFRESH			The frequency at which new LEVEL 0 backups are taken automatically. If NULL, LEVEL 0 backups will not be taken automatically.

26.5 RA_DATABASE_HISTORY

This view lists the storage usage for each protected database.

This view has one row for each database for each day going back one year. There may be rows missing in case the DB_STATS_REFRESH task was delayed in its execution for over a day. There may also be phantom rows for deleted databases.

It is not expected that users will use this data as is. Instead, judicious "GROUP BY" clauses to the data should be applied to obtain data such as:

- average space usage by database per month over the previous year.
- changes in sum(recovery_window_space) across all databases per month over the previous year

Column	Data Type	NULL	Description
DB_UNIQUE_NAME	VARCHAR2(30)		The unique name of this protected database.
DB_KEY	NUMBER	NOT NULL	The primary key for this protected database in the Recovery Appliance.
SAMPLE_TIME	TIMESTAMP(6) WITH TIME ZONE		The day when this database history record was created.
DISK_RESERVED_SPACE	NUMBER		The amount of disk space (in GB) reserved for the exclusive use of this database

Copyright © 2014, 2026, Oracle and/or its affiliates.

Column	Data Type	NULL	Description
SIZE_ESTIMATE	NUMBER		The estimated space (in GB) consumed by the entire protected database.
SPACE_USAGE	NUMBER		The amount of space (in GB) currently used by this protected database.
CUMULATIVE_USAGE	NUMBER		The cumulative amount of disk space (in GB) allocated for all backups received for this database.
REPLICATION_USAGE	NUMBER		The cumulative amount of disk space (in GB) replicated for this protected database.
CLOUD_USAGE	NUMBER		The cumulative amount of disk space (in GB) sent to cloud storage for this protected database.
SBT_USAGE	NUMBER		The cumulative amount of disk space (in GB) sent to SBT from this protected database.
KEEP_SPACE	NUMBER		Amount of KEEP backup space (in GB) currently used by this database.
RECOVERY_WINDOW_SPACE	NUMBER		The estimated space (in GB) that is needed to meet the recovery window goal.
RESTORE_WINDOW	INTERVAL DAY(9) TO SECOND(9)		Time range of backups used to estimate recovery_window_space.
RECOVERY_WINDOW_GOAL	INTERVAL DAY(9) TO SECOND(9)		The recovery window goal for backups on disk, as specified in the protection policy.
MAX_RETENTION_WINDOW	INTERVAL DAY(9) TO SECOND(6)		The maximum amount of time to retain disk backups.
RECOVERY_WINDOW_SBT	INTERVAL DAY(9) TO SECOND(6)		The recovery window for backups on tape, as specified in the protection policy.
CREATION_TIME	TIMESTAMP(6) WITH TIME ZONE		The time when this database was added to the Recovery Appliance.
AUTOTUNE_DISK_RESERVED_SPACE	VARCHAR2(16)		If YES, the Recovery Appliance automatically sets and updates disk_reserved_space as needed. If NO, the administrator of the Recovery Appliance must set and update manually disk_reserved_space.
DB_TIMEZONE	VARCHAR2(16)		The time zone used to create the last backup.
LOW_TIME	DATE		The earliest time to which the protected database can be restored.
HIGH_TIME	DATE		The latest time to which the protected database can be restored.
LAST_UPDATED_RR	DATE		The time that the restore range for this protected database was updated.

26.6 RA_DATABASE_SYNONYM

This view lists the protected databases and their equivalent names.

Column	Data Type	NULL	Description
DB_UNIQUE_NAME	VARCHAR2(512)		The unique name of the protected database.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Column	Data Type	NULL	Description
DBID	NUMBER		The DBID for all protected databases that are equivalent to this database.
TENANT_NAME	VARCHAR2		The tenant name for this database in the Recovery Appliance.
TENANT_IDENTIFIER	NUMBER		The customer tenant identifier for this database in the Recovery Appliance.

26.7 RA_DATABASE_STORAGE_USAGE

This view lists the storage usage for each protected database.

Column	Data Type	NULL	Description
DB_UNIQUE_NAME	VARCHAR2(30)		The unique name of the protected database.
DB_KEY	NUMBER		The primary key for this protected database in the Recovery Appliance metadata database.
STORAGE_LOCATION	VARCHAR2(128)	NOT NULL	The name of the Recovery Appliance storage location used by this protected database.
USED_SPACE	NUMBER		The amount of space (in GB) used by this database in its Recovery Appliance storage locations. Backups for a protected database typically reside in only one storage location, but can reside in two locations when a movement operation is in progress.
TENANT_NAME	VARCHAR2		The tenant name for this database in the Recovery Appliance.
TENANT_IDENTIFIER	NUMBER		The customer tenant identifier for this database in the Recovery Appliance.

26.8 RA_DB_ACCESS

This view describes which Recovery Appliance user accounts have access to which protected databases.

Column	Data Type	NULL	Description
USERNAME	VARCHAR2(128)	NOT NULL	The name of the Recovery Appliance user account.
DB_UNIQUE_NAME	VARCHAR2(32)		The unique name of the protected database accessed by the Recovery Appliance user account.
DB_KEY	NUMBER		The primary key for the protected database accessed by the Recovery Appliance user account.
TENANT_KEY	NUMBER		The tenant key for the protected database accessed by the Recovery Appliance user account.
TENANT_NAME	VARCHAR2		The tenant name for this database in the Recovery Appliance.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Column	Data Type	NULL	Description
TENANT_IDENTIFIER	NUMBER		The customer tenant identifier for this database in the Recovery Appliance.

26.9 RA_DISK_RESTORE_RANGE

The restore range of each protected database from disk backups on this Recovery Appliance.

Column	Data Type	NULL	Description
DB_KEY	NUMBER		The primary key of the protected database.
DB_UNIQUE_NAME	VARCHAR2(30)		The unique name assigned to the database.
TENANT_NAME	VARCHAR2		The tenant name for this database in the Recovery Appliance.
TENANT_IDENTIFIER	NUMBER		The customer tenant identifier for this database in the Recovery Appliance.
LOW_TIME	DATE		The earliest time to which the protected database can be restored.
HIGH_TIME	DATE		The latest time to which the protected database can be restored.
LOW_SCN	NUMBER		The lowest SCN to which the protected database can be restored.
HIGH_SCN	NUMBER		The highest SCN to which the protected database can be restored.
LOW_DBINC_KEY	NUMBER		The primary key for the incarnation of the target database to which LOW_SCN belongs.
HIGH_DBINC_KEY	NUMBER		The primary key for the incarnation of the target database to which HIGH_SCN belongs.
LAST_UPDATED	DATE		The time that the restore range for this protected database was updated.
KEEP_OPTIONS	NUMBER		If 0, then this is a row with non-keep backups. If >0, this row has keep backups.
KEEP_UNTIL	DATE		Contains the retention time for keep backups. NULL means the restore range consists of a KEEP FOREVER backup. Valid only for rows with KEEP_OPTIONS > 0.
KEEPBKP_TAG	NUMBER		Contains the tag value for the archival backup. Valid only for rows with KEEP_OPTIONS > 0.
KEEPBKP_RSPNAME	VARCHAR2		Contains the restore point name of the archival backup. Valid only for rows with KEEP_OPTIONS > 0.
KEEPBKP_TO_SCN	NUMBER		Contains the recovery SCN for which the archival backups were creaed. Valid only for rows with KEEP_OPTIONS > 0.
KEEPBKP_TO_TIME	DATE		Contains the recovery time for which the archival backups were created. Valid only for rows with KEEP_OPTIONS > 0.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Column	Data Type	NULL	Description
BACKUP_TAG	NUMBER		Contains the tag value of backups for the range. If the tag value of all backups is not the same, it contains *'''.

26.10 RA_EM_SBT_JOB_TEMPLATE

This view lists defined SBT jobs and their statuses for Oracle Enterprise Manager.

Column	Data Type	NULL	Description
TEMPLATE_NAME	VARCHAR2(128)	NOT NULL	The name of the SBT job template.
FULL_TEMPLATE_NAME	VARCHAR2(128)	NOT NULL	The full name of the SBT job template.
POLICY_NAME	VARCHAR2(128)		The protection policy specifying the protected databases whose backups the Recovery Appliance considers eligible for copying.
DB_UNIQUE_NAME	VARCHAR2(512)		The unique name of the protected database whose backups the Recovery Appliance considers eligible for copying.
ATTRIBUTE_SET_NAME	VARCHAR2(128)	NOT NULL	The name of the SBT attribute set.
LIB_NAME	VARCHAR2(128)	NOT NULL	The name of the SBT library.
BACKUP_TYPE	VARCHAR2(16)		The types of backups to be copied to tape by this job: ALL, FULL, INCR, ARCH, or TAPE_RESERVE.
PRIORITY	NUMBER		The priority for scheduling this job.
COPIES	NUMBER	NOT NULL	The number of copies to be created on tape.
WINDOW	INTERVAL DAY(2) TO SECOND(6)		The time allotted for copy tasks to start for this job.
FROM_TAG	VARCHAR2(32)		The tag for the backup to be copied to tape by this job.
ERROR_TEXT	VARCHAR2(4000)		The error text for the task that failed.
ERROR_LAST_SEEN	TIMESTAMP(6) WITH TIME ZONE		The timestamp when the Recovery Appliance most recently detected the error.
EXECUTABLE	NUMBER		The number of tasks in an executable state.
RUNNING	NUMBER		The number of tasks that are running or retrying.
COMPLETED	NUMBER		The number of completed tasks.
COMPLETION_TIME	TIMESTAMP(6) WITH TIME ZONE		The time of the most recent completed task.
STATUS	VARCHAR2(5)		The status of the SBT library: READY, PAUSE, or ERROR.
BYTES	NUMBER		The number of bytes read or written so far.
COMPRESSION_ALGORITHM	VARCHAR2(6)		The compression algorithm used by this job: NONE, BASIC, LOW, MEDIUM, or HIGH. Note that this column is available only with Zero Data Loss Recovery Appliance software update 12.1.1.1.8 and later.

Copyright © 2014, 2026, Oracle and/or its affiliates.

26.11 RA_ENCRYPTION_INFO

This view describes the historical encryption key information.

Column	Data Type	NULL	Description
ENCINFO_KEY	NUMBER	NOT NULL	The key of this encryption info record in the Recovery Appliance metadata database.
DB_UNIQUE_NAME	VARCHAR2(32)	NOT NULL	The unique name of the database for this encryption info record..
DBID	NUMBER	NOT NULL	
CREATE_TIME	TIMESTAMP(6) WITH TIME ZONE		The time at which the key was created..
ENCRYPTION_TAG	VARCHAR2(128)	NOT NULL	The tag associated with this encryption key..
ENCRYPTION_KEYID	VARCHAR2(78)	NOT NULL	The encryption id associated with this encryption key.
TENANT_NAME	VARCHAR2		The tenant name for this database in the Recovery Appliance.
TENANT_IDENTIFIER	NUMBER		The customer tenant identifier for this database in the Recovery Appliance.

26.12 RA_INCIDENT_LOG

This view describes the Recovery Appliance incidents.

Column	Data Type	NULL	Description
INCIDENT_ID	NUMBER		The unique ID for the incident.
ERROR_CODE	NUMBER		The Oracle error code for the message describing the incident.
PARAMETER	VARCHAR2(1000)		The parameter qualifying the scope of the error code.
ERROR_TEXT	VARCHAR2(4000)		The text of the message for the last detection of this error condition.
SL_KEY	NUMBER		Primary key of the storage location (if any) involved in this incident
SL_NAME	VARCHAR2(128)		The primary key of the Recovery Appliance storage location (if any) involved in this incident.
DB_KEY	NUMBER		The primary key of the protected database (if any) involved in this incident.
TENANT_KEY	NUMBER		The tenant key for the protected database accessed by the Recovery Appliance user account.
DB_UNIQUE_NAME	VARCHAR2(30)		The unique name of the protected database (if any) involved in this incident.
TASK_ID	NUMBER		The ID for the task (if any) in which this incident was detected.
STATUS	VARCHAR2(6)		The status of this incident: ACTIVE, FIXED, or RESET.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Column	Data Type	NULL	Description
COMPONENT	VARCHAR2(30)		The component of the Recovery Appliance detecting this incident.
SEVERITY	VARCHAR2(47)		The importance of this incident to the smooth operation of the Recovery Appliance.
FIRST_SEEN	TIMESTAMP(6) WITH TIME ZONE	NOT NULL	The timestamp when the Recovery Appliance first detected the incident.
LAST_SEEN	TIMESTAMP(6) WITH TIME ZONE	NOT NULL	The timestamp when the Recovery Appliance most recently detected the incident.
SEEN_COUNT	NUMBER	NOT NULL	The number of times that the Recovery Appliance detected the incident.
CALL_STACK	VARCHAR2(4000)		The call stack at the time of the incident.

26.13 RA_INCOMING_BACKUP_PIECES

This view describes the backup pieces being received by the Recovery Appliance.

Column	Data Type	NULL	Description
SL_KEY	NUMBER		The primary key of the Recovery Appliance storage location storing this backup piece.
SL_NAME	VARCHAR2(128)		The name of the Recovery Appliance storage location storing this backup piece.
DB_KEY	NUMBER		The primary key of the protected database creating this backup piece.
DB_UNIQUE_NAME	VARCHAR2(30)		The unique name of the protected database creating this backup piece.
HANDLE	VARCHAR2(1024)		The handle assigned to this backup piece.
CURRENT_SIZE	NUMBER		The size (in GB) currently allocated for this backup piece.
START_TIME	TIMESTAMP(6) WITH TIME ZONE		The time when the backup piece was first seen by the Recovery Appliance.
LAST_UPDATE	TIMESTAMP(6) WITH TIME ZONE		The time when the backup piece was completely received.
REP_PAIR_KEY	NUMBER		Join with RA_REPLICATION_CONFIG to determine sending upstream Recovery Appliance.
TENANT_NAME	VARCHAR2		The tenant name for this database in the Recovery Appliance.
TENANT_IDENTIFIER	NUMBER		The customer tenant identifier for this database in the Recovery Appliance.

26.14 RA_POLLING_FILES

This view describes the set of files the Recovery Appliance backed up from the configured polling location.

Note that this view is available only with Zero Data Loss Recovery Appliance software update 12.1.1.1.8 and later.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Column	Data Type	NULL	Description
POLL_NAME	VARCHAR2(128)		The name of the polling policy.
POLL_KEY	NUMBER	NOT NULL	The unique identifier of the polling policy.
FILE_NAME	VARCHAR2(512)		The name of the file discovered in the configured polling location.
FILE_SIZE	NUMBER		The size of the file the last time it was scanned in the polling location.
STATUS	VARCHAR2(24)		The current state of processing of the file. PROCESSING: file has just been recognized by Recovery Appliance as a candidate file. COMPLETED: file was accepted by the Recovery Appliance. DBID NOT IN POLICY database that created this file is not associate with the polling policy that is processing the file. UNKNOWN FILE TYPE file type of file is not supported by the Recovery Appliance. ERROR IN PROCESSING FILE an error was detected in processing the file. Check ra_incident_log for details. INCOMPLETE FILE: file was never completed written by source database. FILE DELETED BY ZDLRA: processing was completed and file was removed per polling policy.
LAST_PROCESSED	TIMESTAMP WITH TIME ZONE		The time stamp includes the time zone of when last processed..

26.15 RA_POLLING_POLICY

This view lists the defined backup polling policies.

Column	Data Type	NULL	Description
POLLING_NAME	VARCHAR2(128)		The name of this backup polling policy.
POLLING_KEY	NUMBER	NOT NULL	The primary key for this backup polling policy in the recovery catalog.
DEST	VARCHAR2(4000)	NOT NULL	The file system directory corresponding to the backup polling location.
FREQUENCY	INTERVAL DAY(9) TO SECOND(6)		The interval at which the Recovery Appliance scans the backup polling location for new files.
NEXT_EXECUTE	TIMESTAMP(6) WITH TIME ZONE		The next time when the polling location will be scanned.
DELETE_INPUT	VARCHAR2(5)		The deletion policy for the polling location: TRUE to delete files as they are accepted; FALSE to keep all files.

Copyright © 2014, 2026, Oracle and/or its affiliates.

26.16 RA_PROTECTION_POLICY

This view lists the protection policies defined for this Recovery Appliance.

Column	Data Type	NULL	Description
POLICY_NAME	VARCHAR2(128)	NOT NULL	The user-created name of the protection policy.
DESCRIPTION	VARCHAR2(128)		The protection policy description.
PROT_KEY	NUMBER	NOT NULL	The primary key for this protection policy in the Recovery Appliance metadata database.
SL_NAME	VARCHAR2(128)	NOT NULL	The name of the Recovery Appliance storage location used by this protection policy.
SL_KEY	NUMBER	NOT NULL	The primary key of the Recovery Appliance storage location used by this protection policy.
POLLING_NAME	VARCHAR2(128)		The name of the backup polling policy assigned to this protection policy.
RECOVERY_WINDOW_GOAL	INTERVAL DAY(9) TO SECOND(6)		The recovery window goal for restoring backups stored on disk.
MAX_RETENTION_WINDOW	INTERVAL DAY(9) TO SECOND(6)		The maximum amount of time that the Recovery Appliance must retain disk backups.
RECOVERY_WINDOW_SBT	INTERVAL DAY(9) TO SECOND(6)		The recovery window for restoring backups stored on tape.
UNPROTECTED_WINDOW	INTERVAL DAY(9) TO SECOND(6)		The point beyond which recovery is not possible unless additional redo is available.
GUARANTEED_COPY	VARCHAR2(3)		The status of the guaranteed copy setting: YES means that the Recovery Appliance replicates backups or copies them to tape before deleting them; NO means that the Recovery Appliance accepts new backups even if old backups must be purged because free space is low.
REPLICATION_SERVER_LIST	VARCHAR2(4000)		The list of replication server configurations associated with this protection policy.
ALLOW_BACKUP_DELETION	VARCHAR2(3)		The setting that controls whether RMAN backups for databases that use this protection policy can be deleted: NO means that the Recovery Appliance does not allow deletion of these backups; YES means that the Recovery Appliance allows deletion of these backups. Note that this parameter is available only with Zero Data Loss Recovery Appliance software update 12.1.1.1.7 and later.
STORE_AND_FORWARD	VARCHAR2(3)		The status of the Backup and Redo Failover setting: YES means that the Recovery Appliance applies the Backup and Redo Failover strategy to backups from the databases associated with this protection policy; NO means that the Backup and Redo Failover feature is not enabled and the Recovery Appliance applies the normal incremental-forever backup strategy instead. Note that this setting is available only with Zero Data Loss Recovery Appliance software update 12.1.1.1.8 and later.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Column	Data Type	NULL	Description
AUTOTUNE_RESERVED_SPACE	VARCHAR2(3)		YES: Recovery Appliance automatically sets and updates disk_reserved_space as needed. NO: The administrator of the Recovery Appliance must set and update disk_reserved_space manually.
LOG_COMPRESSION_ALGORITHM	VARCHAR2(6)	NOT NULL	The status of the log compression algorithm setting: HIGH means that the Recovery Appliance uses RMAN HIGH compression for compressing archive logs; BASIC means that the Recovery Appliance uses RMAN BASIC compression for compressing archive logs; LOW means that the Recovery Appliance uses RMAN LOW compression for compressing archive logs; MEDIUM means that the Recovery Appliance uses RMAN MEDIUM compression for compressing archive logs; OFF means that the Recovery Appliance does not compress the archive logs.
RECOVERY_WINDOW_COMPLIANCE	INTERVAL DAY(9) TO SECOND(6)		Time interval during which backups may not be deleted after they are created.
KEEP_COMPLIANCE	VARCHAR2(3)	NOT NULL	YES: The Recovery Appliance will prevent the deletion of KEEP backups until their KEEP_UNTIL expires. The Recovery Appliance also prevents RMAN from changing the keep time for backups. NO: The administrator of the Recovery Appliance is permitted to remove KEEP backups. RMAN can change the keep time for backups.

26.17 RA_PURGING_QUEUE

This view describes the order in which protected databases will have their oldest backups deleted when space is low.

Column	Data Type	NULL	Description
SL_NAME	VARCHAR2(128)	NOT NULL	The Recovery Appliance storage location name.
SL_KEY	NUMBER		The primary key for this Recovery Appliance storage location in the recovery catalog.
DB_UNIQUE_NAME	VARCHAR2(30)		The unique name of the protected database whose backups the Recovery Appliance will purge.
DB_KEY	NUMBER	NOT NULL	The primary key for the protected database whose backups the Recovery Appliance will purge.
PURGE_ORDER	NUMBER		The order in which this protected database is eligible for purging.
NEW_RECOVERY_WINDOW	INTERVAL DAY(9) TO SECOND(6)		The recovery window goal for this protected database after a purge.
NEW_PCT_RECOVERY	NUMBER		The percentage of the recovery window goal remaining for this protected database after a purge.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Column	Data Type	NULL	Description
PCT_STORAGE	NUMBER		The percentage of reserved space being consumed by this protected database.
TENANT_NAME	VARCHAR2		The tenant name for this database in the Recovery Appliance.
TENANT_IDENTIFIER	NUMBER		The customer tenant identifier for this database in the Recovery Appliance.

26.18 RA_RECOVERY_COMPLIANCE

This view lists information about active compliance windows on the database in which backups may not be deleted..

Column	Data Type	NULL	Description
DB_UNIQUE_NAME	VARCHAR2(30)		The unique database name.
DB_KEY	NUMBER	NOT NULL	The primary key for this database in the recovery catalog.
DURATION	INTERVAL DAY(9) TO SECOND(9)	NOT NULL	Required retention period for backups created while this compliance policy is active.
START_TIME	TIMESTAMP(6) WITH TIME ZONE	NOT NULL	Starting time when this compliance policy goes into effect.
END_TIME	TIMESTAMP(6) WITH TIME ZONE		Ending time when this compliance policy expires. NULL means that this compliance policy is still active.
TENANT_NAME	VARCHAR2		The tenant name for this database in the Recovery Appliance.
TENANT_IDENTIFIER	NUMBER		The customer tenant identifier for this database in the Recovery Appliance.

26.19 RA_REPLICATION_CONFIG

This view lists the replication server configurations.

Column	Data Type	NULL	Description
REPLICATION_SERVER_NAME	VARCHAR2(128)	NOT NULL	The user-assigned name of the replication server configuration.
REPLICATION_SERVER_STATE	VARCHAR2(21)		The replication status of the downstream Recovery Appliance: AVAILABLE, CREATING, DELETING, TESTING COMMUNICATION, RUNNING, PAUSED, ERROR STATE or null.
REPLICATION_SERVER_ROLE	VARCHAR2(10)		The user-assigned name of the replication server configuration.
CATALOG_OWNER	VARCHAR2(128)	NOT NULL	The owner of the Recovery Catalog schema that will be connecting to the DS Recovery Appliance.
PROXY_HTTP_ADDRESS	VARCHAR2(519)		The address of the proxy server in the form proxy_server_http_address:port_of_proxy_server.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Column	Data Type	NULL	Description
PROXY_TIMEOUT	NUMBER		The timeout period (in seconds) for the proxy server connection.
SBT_LIBRARY_NAME	VARCHAR2(128)	NOT NULL	The name of the SBT library with which this replication server configuration is associated.
SBT_LIBRARY_PARMS	VARCHAR2(1024)		The SBT library parameters.
ATTRIBUTE_NAME	VARCHAR2(128)	NOT NULL	The SBT attribute set name.
ATTRIBUTE_PARMS	VARCHAR2(1024)		The SBT parameters passed while allocating the RMAN channel.
WALLET_PATH	VARCHAR2(512)		The path to the local Oracle wallet (excluding the wallet file name).
WALLET_ALIAS	VARCHAR2(512)	NOT NULL	The alias that identifies the Oracle wallet credentials that this Recovery Appliance uses to log in to the downstream Recovery Appliance.
SERVER_HOST	CLOB	NOT NULL	The server name or address of the downstream Recovery Appliance.
MAX_STREAMS	NUMBER	NOT NULL	The maximum number of simultaneous replication tasks. Each replication task processes a single backup piece.
PAIRED_REPLICATION_SERVER	VARCHAR2(3)		YES if this replication server is paired with another replication server. Examine RA_REPLICATION_PAIR for additional information.
REP_PAIR_KEY	NUMBER		Cross reference to RA_REPLICATION_PAIR view
REPLICATION_SERVER_KEY	NUMBER		Cross reference to associated RA_REPLICATION_PAIR_* views

26.20 RA_REPLICATION_DATABASE

This view lists information on replication servers and protected databases.

Column	Data Type	NULL	Description
DB_UNIQUE_NAME	VARCHAR2(128)		The unique name of the protected database.
DB_KEY	NUMBER	NOT NULL	The primary key for this protected database in the Recovery Appliance metadata database.
REPLICATION_SERVER_NAME	VARCHAR2(128)	NOT NULL	The user-assigned name of the replication server.
REPLICATION_SERVER_KEY	NUMBER	NOT NULL	Cross reference to other RA_* views and server.server_key.
REPLICATION_SERVER_STATE	VARCHAR2(18)		The possible replication server status are: • AVAILABLE • CREATING • DELETING • TESTING CONNECTION • RUNNING • PAUSED • ERROR STATE

Copyright © 2014, 2026, Oracle and/or its affiliates.

Column	Data Type	NULL	Description
REPLICATION_DATABASE_STATE	VARCHAR2(7)	NOT NULL	The possible database replication status are: • AVAILABLE • CREATING • DELETING • TESTING CONNECTION • RUNNING • PAUSED • ERROR STATE • null
POLICY_NAME	VARCHAR2(128)	NOT NULL	The protection policy that is associated with this replication server and database.
FAILURE_CNT	NUMBER	NOT NULL	The number of conseceutive failures this database has had replicating to this replication server.
LAST_RECONCILE	TIMESTAMP(6) WITH TIME ZONE		The time of the last successful reconcile for this database and replication server.
NEXT_RECONCILE	TIMESTAMP(6) WITH TIME ZONE		The time of the next reconcile for this database and replication server.
LAST_REPLICATION	TIMESTAMP(6) WITH TIME ZONE		The time of the last successful replication for this database and replication server.
PENDING_REPLICATION_SETUP	VARCHAR2(3)		If YES, an initial connection, reconcile and replication is still pending. If NO, initial replication setup is complete.
FIXED_SECTION_UPDATE	TIMESTAMP(6) WITH TIME ZONE		The time of the most recent fixed section update from the client database that has not yet been reconciled to this replication server.
REP_RSDB_KEY	NUMBER	NOT NULL	Cross reference to other RA_* views
REP_SERVER_KEY	NUMBER	NOT NULL	Cross reference to other RA_* views

26.21 RA_REPLICATION_PAIR

This view lists replication information for replicating protection policies.

Column	Data Type	NULL	Description
REPLICATION_SERVER_NAME	VARCHAR2(128)	NOT NULL	The user-assigned name of the replication server configuration. If we are an upstream to this remote Recovery Appliance, the user-assigned name of the replication server.
REPLICATION_SERVER_STATE	VARCHAR2(21)		The replication status of the downstream Recovery Appliance: AVAILABLE, CREATING, DELETING, TESTING COMMUNICATION, RUNNING, PAUSED, ERROR STATE or null.
REPLICATION_SERVER_ROLE	VARCHAR2(10)		shows the role of this Recovery Appliance with associated with another Recovery Appliance: UPSTREAM, DOWNSTREAM, or PAIRED.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Column	Data Type	NULL	Description
REMOTE_RA_DBID	NUMBER		Information from the downstream Recovery Appliance. The v$database.dbid of the downstream Recovery Appliance.
REMOTE_RA_DBUNAME	VARCHAR2(30)		Information from the downstream Recovery Appliance. The v$database.name of the downstream Recovery Appliance.
REMOTE_CONNECTING_USERNAME	VARCHAR2(128)		Incoming replication user from the downstream Recovery Appliance.
REMOTE_CONNECTING_USER_ID	NUMBER		Incoming replication user_id from the downstream Recovery Appliance.
REMOTE_VERSION	VARCHAR2(256)		Information from the downstream Recovery Appliance. ZDLRA software version string from sys.raa_versions.release_version.
REMOTE_CVERSION	VARCHAR2(256)		Information from the downstream Recovery Appliance. Comparable ZDLRA software version string from sys.raa_versions.release_version.
REMOTE_BUILD_STRING	VARCHAR2(100)		Information from the downstream Recovery Appliance config._build string.
REMOTE_VERSION_RECONCILE	NUMBER	NOT NULL	Cross reference to associated RA_REPLICATION_* views.
REPLICATION_SERVER_KEY	NUMBER		Cross reference to associated RA_REPLICATION_* views

26.22 RA_REPLICATION_POLICY

This view lists the association of replication servers to protection policy.

Column	Data Type	NULL	Description
REPLICATION_SERVER_NAME	VARCHAR2(128)	NOT NULL	The user-assigned name of the replication server configuration.
REPLICATION_SERVER_STATE	VARCHAR2(21)		The replication status of the downstream Recovery Appliance: AVAILABLE, CREATING, DELETING, TESTING CONNECTION, RUNNING, PAUSED, ERROR STATE or null.
POLICY_NAME	VARCHAR2(128)		The protection policy associated with this replication server configuration.
READ_ONLY	VARCHAR2(3)		YES: The downstream Recovery Appliance is treated as a read-only device. Backups can be retrieved from the downstream but are not replicated.
REQUEST_ONLY	VARCHAR2(3)		YES: The downstream Recovery Appliance is treated as a request-only device. REQUEST_ONLY implies READ_ONLY, but only the requested backups are sent from the downstream.
REPLICATION_SERVER_KEY	NUMBER	NOT NULL	Cross reference to other RA_REPLICATION_* views

Copyright © 2014, 2026, Oracle and/or its affiliates.

Column	Data Type	NULL	Description
STORE_AND_FORWARD	VARCHAR2(3)		YES when Recovery Appliance is being used to store backups temporarily until they can be moved downstream. NO for normal disaster recovery support.

26.23 RA_RESTORE_RANGE

This view describes the restore range of each protected database from all backups on this Recovery Appliance.

Column	Data Type	NULL	Description
DB_KEY	NUMBER		The primary key of the protected database.
DB_UNIQUE_NAME	VARCHAR2(30)		The unique name assigned to the database.
TENANT_NAME	VARCHAR2		The tenant name for this database in the Recovery Appliance.
TENANT_IDENTIFIER	NUMBER		The customer tenant identifier for this database in the Recovery Appliance.
LOW_TIME	DATE		The earliest time to which the protected database can be restored.
HIGH_TIME	DATE		The latest time to which the protected database can be restored.
LOW_SCN	NUMBER		The lowest SCN to which the database can be restored.
HIGH_SCN	NUMBER		The highest SCN to which the protected database can be restored.
LOW_DBINC_KEY	NUMBER		The primary key for the incarnation of the target database to which the low SCN belongs.
HIGH_DBINC_KEY	NUMBER		The primary key for the incarnation of the target database to which the high SCN belongs.
LAST_UPDATED	DATE		The time that the restore range for this database was updated.
KEEP_OPTIONS	NUMBER		If 0, then this is a row with non-keep backups. If >0, this row has keep backups.
KEEP_UNTIL	DATE		Contains the retention time for keep backups. NULL means the restore range consists of a KEEP FOREVER backup. Valid only for rows with KEEP_OPTIONS > 0.
KEEPBKP_TAG	NUMBER		Contains the tag value for the archival backup. Valid only for rows with KEEP_OPTIONS > 0.
KEEPBKP_RSPNAME	VARCHAR2		Contains the restore point name of the archival backup. Valid only for rows with KEEP_OPTIONS > 0.
KEEPBKP_TO_SCN	NUMBER		Contains the recovery SCN for which the archival backups were creaed. Valid only for rows with KEEP_OPTIONS > 0.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Column	Data Type	NULL	Description
KEEPBKP_TO_TIME	DATE		Contains the recovery time for which the archival backups were created. Valid only for rows with KEEP_OPTIONS > 0.
BACKUP_TAG	NUMBER		Contains the tag value of backups for the range. If the tag value of all backups is not the same, it contains *'''.

26.24 RA_REQUEST_BACKUP

This view describes the list of requested backup pieces from a remote Recovery Appliance.

Column	Data Type	NULL	Description
RRA_REQUEST_BACKUP_KEY	NUMBER	NOT NULL	The key of this request record.
GROUP#	NUMBER		All backups with this group# are part of a single request group.
BATCH_ID	NUMBER		All backups with this batch_id were requested at the same time.
REP_PAIR_KEY	NUMBER		The key to the replication pair record.
BP_KEY	NUMBER		The local bp_key of the bp record being requested.
DB_KEY	NUMBER		The local db_key of the owner of the backup piece being requested.
HANDLE	VARCHAR2(1024)		The handle of the backup piece being requested.
SET_STAMP	NUMBER		The set_stamp of the backup piece being requested.
SET_COUNT	NUMBER		The set_count of the backup piece being requested.
PIECE#	NUMBER		The piece# of the backup piece being requested.
CREATE_TIME	TIMESTAMP(6) WITH TIME ZONE		The time at which the backup request record was created.
LAST_STATUS_UPDATE	TIMESTAMP(6) WITH TIME ZONE		The most recent time at which the backup request record was updated.
STATUS	VARCHAR2(13)		The most recent status of the backup request record.
DB_UNIQUE_NAME	VARCHAR2(32)	NOT NULL	The unique name of the database for this encryption info record..
TENANT_NAME	VARCHAR2		The tenant name for this database in the Recovery Appliance.
TENANT_IDENTIFIER	NUMBER		The customer tenant identifier for this database in the Recovery Appliance.

26.25 RA_SBT_ATTRIBUTE_SET

This view describes the defined SBT attribute set.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Column	Data Type	NULL	Description
ATTRIBUTE_SET_KEY	NUMBER	NOT NULL	The key of this SBT attribute set in the Recovery Appliance metadata database.
ATTRIBUTE_SET_NAME	VARCHAR2(128)	NOT NULL	The SBT attribute set name.
LIB_NAME	VARCHAR2(128)	NOT NULL	The name of the SBT library object with which this attribute set is associated.
STREAMS	NUMBER		The number of parallel streams available for jobs that run with this attribute set.
POOLID	NUMBER		The media pool identifier.
PARMS	VARCHAR2(1024)		The SBT parameters passed while allocating the RMAN channel.
SEND	VARCHAR2(1024)		The SEND command string passed to the allocated channel.

26.26 RA_SBT_JOB

This view describes the defined SBT job templates.

Column	Data Type	NULL	Description
TEMPLATE_KEY	NUMBER	NOT NULL	The key of this SBT job template in the Recovery Appliance metadata database.
TEMPLATE_NAME	VARCHAR2(128)	NOT NULL	The SBT job template name.
ATTRIBUTE_SET_NAME	VARCHAR2(128)	NOT NULL	The SBT attribute set name.
LIB_NAME	VARCHAR2(128)	NOT NULL	The SBT library name.
POLICY_NAME	VARCHAR2(128)		The protection policy specifying databases whose backups the Recovery Appliance considers eligible for copying to tape.
DB_KEY	NUMBER		The primary key of the protected database whose backups the Recovery Appliance considers eligible for copying to tape.
DB_UNIQUE_NAME	VARCHAR2(30)		The unique name of the protected database whose backups the Recovery Appliance considers eligible for copying to tape.
BACKUP_TYPE	VARCHAR2(16)		The types of backups to be copied to tape by this job: ALL, FULL, INCR, ARCH, or TAPE_RESERVE.
FROM_TAG	VARCHAR2(32)		The backups with the specified tag to be copied to tape by this job.
PRIORITY	NUMBER		The priority for scheduling this job.
COPIES	NUMBER	NOT NULL	The number of copies to be created on tape.
LAST_SCHEDULE_TIME	TIMESTAMP(6) WITH TIME ZONE		The last time at which this SBT job was scheduled to run.
WINDOW	INTERVAL DAY(2) TO SECOND(6)		The time allotted for copy tasks to start for this job.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Column	Data Type	NULL	Description
COMPRESSION_ALGORITHM	VARCHAR2(6)		The compression algorithm used by this job: NONE, BASIC, LOW, MEDIUM, HIGH, or OFF. OFF indicates that compression was explicitly turned off for this job (the compression setting in the SBT job template was ignored). Note that this column is available only with Zero Data Loss Recovery Appliance software update 12.1.1.1.8 and later.
TENANT_NAME	VARCHAR2		The tenant name for this database in the Recovery Appliance.
TENANT_IDENTIFIER	NUMBER		The customer tenant identifier for this database in the Recovery Appliance.

26.27 RA_SBT_LIBRARY

This view lists the defined SBT libraries.

Column	Data Type	NULL	Description
LIB_KEY	NUMBER	NOT NULL	The key of this SBT library in the Recovery Appliance metadata database.
LIB_NAME	VARCHAR2(128)	NOT NULL	The SBT library name.
DRIVES	NUMBER	NOT NULL	The number of drives available for use by this SBT library.
RESTORE_DRIVES	NUMBER	NOT NULL	The number of drives reserved for restore operations.
PARMS	VARCHAR2(1024)		The SBT parameters passed while allocating an RMAN channel.
SEND	VARCHAR2(1024)		The SEND command string passed to the allocated channel.
REQUIRES_ENCRYPTION	VARCHAR2(3)	NOT NULL	This library requires backups to be sent out encrypted.
STATUS	VARCHAR2(5)		The SBT library status: READY, PAUSE, ERROR, or null.
LIBTYPE	VARCHAR2(11)		Specifies the type of the library. Values are: CLOUD, TAPE, and REPLICATION
GUARANTEED	VARCHAR2(3)	NOT NULL	YES: Library may be used as a backing store for GUARANTEED COPY. NO Library may not be used as a backing store for the GUARANTEED COPY.
IMMUTABLE	VARCHAR2(3)	NOT NULL	YES: Library may be used as a backing store for KEEP_COMPLIANCE. NO: Library may not be used as a backing store for the KEEP_COMPLIANCE.
LAST_ERROR_TEXT	VARCHAR2(4000)		The most recent error text of the task that failed.

Copyright © 2014, 2026, Oracle and/or its affiliates.

26.28 RA_SBT_MIRROR

This view lists the SBT libraries that are being mirrored with remote Recovery Appliances.

Column	Data Type	NULL	Description
LIB_KEY	NUMBER	NOT NULL	The local SBT library key.
LIB_NAME	VARCHAR2(128)		The local SBT library name.
LIB_TYPE	VARCHAR2(5)		Specifies the type of library.
REMOVE_LIB_KEY	NUMBER		The SBT library key of mirrored library on remote Recovery Appliance.
REMOTE_PAIR_KEY	NUMBER		Join with RA_REPLICATION_PAIR to determine the remote Recovery Appliance where the mirrored library exists.

26.29 RA_SBT_RESTORE_RANGE

This view describes the restore range of each database from SBT backups on the Recovery Appliance.

Column	Data Type	NULL	Description
DB_KEY	NUMBER		The primary key of the protected database.
DB_UNIQUE_NAME	VARCHAR2(30)		The unique name assigned to the database.
TENANT_NAME	VARCHAR2		The tenant name for this database in the Recovery Appliance.
TENANT_IDENTIFIER	NUMBER		The customer tenant identifier for this database in the Recovery Appliance.
LOW_TIME	DATE		The earliest time to which the database can be restored.
HIGH_TIME	DATE		The latest time to which the database can be restored.
LOW_SCN	NUMBER		The lowest SCN to which the database can be restored.
HIGH_SCN	NUMBER		The highest SCN to which the database can be restored.
LOW_DBINC_KEY	NUMBER		The primary key for the incarnation of the target database to which the low SCN belongs.
HIGH_DBINC_KEY	NUMBER		The primary key for the incarnation of the target database to which the high SCN belongs.
LAST_UPDATED	DATE		The time that the restore range for this protected database was last updated.
KEEP_OPTIONS	NUMBER		If 0, then this is a row with non-keep backups. If >0, this row has keep backups.
KEEP_UNTIL	DATE		Contains the retention time for keep backups. NULL means the restore range consists of a KEEP FOREVER backup. Valid only for rows with KEEP_OPTIONS > 0.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Column	Data Type	NULL	Description
KEEPBKP_TAG	NUMBER		Contains the tag value for the archival backup. Valid only for rows with KEEP_OPTIONS > 0.
KEEPBKP_RSPNAME	VARCHAR2		Contains the restore point name of the archival backup. Valid only for rows with KEEP_OPTIONS > 0.
KEEPBKP_TO_SCN	NUMBER		Contains the recovery SCN for which the archival backups were creaed. Valid only for rows with KEEP_OPTIONS > 0.
KEEPBKP_TO_TIME	DATE		Contains the recovery time for which the archival backups were created. Valid only for rows with KEEP_OPTIONS > 0.
BACKUP_TAG	NUMBER		Contains the tag value of backups for the range. If the tag value of all backups is not the same, it contains *''.

26.30 RA_SBT_TASK

This view lists the queued background SBT tasks and their run statuses.

Column	Data Type	NULL	Description
TASK_ID	NUMBER		The ID for the task.
STATE	VARCHAR2(47)		The processing state for the task: EXECUTABLE, RUNNING, COMPLETED, TASK_WAIT, FAILED, and so on.
COMPLETION_TIME	TIMESTAMP(6) WITH TIME ZONE		The timestamp for task completion. The column is null if the task is not complete.
ELAPSED_SECONDS	NUMBER		The elapsed run time (in seconds) for the task.
EXECUTE_INSTANCE_ID	NUMBER		The ID of the database instance ID on which the task must run. The column is null if the task can run on any instance.
ERROR_COUNT	NUMBER		The number of times that the task had errors.
ERROR_TEXT	VARCHAR2(4000)		The error text for the task that failed.
DB_UNIQUE_NAME	VARCHAR2(30)		The unique name of the protected database for which the task is running.
DB_KEY	NUMBER		The primary key of the protected database for which the task is running.
RESTORE_TASK	VARCHAR2(3)		The type of task: YES if this is a restore task; NO if this is a backup task.
BS_KEY	NUMBER		The key of the backup set that is accessed by this task.
PIECE#	NUMBER		The number of the backup piece that is accessed by this task.
COPIES	NUMBER		The number of copies created by this task.
TEMPLATE_NAME	VARCHAR2(128)		The SBT job template to which this task belongs.
ATTRIBUTE_SET_NAME	VARCHAR2(128)		The name of the SBT attribute set to which this task belongs.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Column	Data Type	NULL	Description
LIB_NAME	VARCHAR2(128)	NOT NULL	The name of the SBT library used by this task.
COMPRESSION_ALGORITHM	VARCHAR2(6)		The compression algorithm used by this task: NONE, BASIC, LOW, MEDIUM, HIGH, or OFF. OFF indicates that compression was explicitly turned off for this task (the compression setting in the SBT job template was ignored).
ENCRYPTION_ALGORITHM	VARCHAR2(6)		Encryption algorithm used by this task.
REPLICATION	VARCHAR2(3)		The type of task: YES if this is a replication task; NO if this is an SBT task.
FILENAME	VARCHAR2(513)		The name of the backup file being read or written.
START_TIME	TIMESTAMP(6) WITH TIME ZONE		The start time of this task.
BYTES	NUMBER		The number of bytes read or written so far.
TOTAL	NUMBER		The total number of bytes to be read or written.
IS_GROUP_BACKUP	VARCHAR2(3)		YES means that the backup was created by assembling a set of individual archive logs into a bigger piece to improve tape performance. NO means that the backup was created by individual archive logs.
LIB_KEY	NUMBER	NOT NULL	Key for tape/replication library used by task.
STATUS	VARCHAR2(5)		The SBT library status: READY, PAUSE, ERROR, or null.
LIB_TYPE	VARCHAR2(11)		Specifies the type of library.
ARCHIVED	CHAR(1)		N if this is a current task, Y if this is a completed task.
KEEP_OPTIONS	NUMBER		The keep options of the backup set that is accessed by this task
KEEP_UNTIL	TIMESTAMP(6) WITH TIME ZONE		The retention time of the backup set that is accessed by this task.
TAG	VARCHAR2(128)		The user specified tag for copied piece that is accessed by this task.

26.31 RA_SBT_TEMPLATE_MDF

This view lists missing level 0 data file backups for each SBT template.

Column	Data Type	NULL	Description
TEMPLATE_KEY	NUMBER	NOT NULL	The key identifying the SBT template.
DB_KEY	NUMBER	NOT NULL	The key for the protected database that contains the missing file.
DB_UNIQUE_NAME	VARCHAR2(30)		The unique name of the database that contains the missing data file.
DF_FILE#	NUMBER		The number of the missing data file.
DF_TS#	NUMBER		The tablespace number of the missing data file.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Column	Data Type	NULL	Description
DF_PLUGIN_CHANGE#	NUMBER		The plugin SCN for the missing data file.
DF_FOREIGN_DBID	NUMBER		The foreign DBID for the database that contains the missing data file.
DF_TABLESPACE	VARCHAR2(30)		The tablespace that contains the missing data file.
DF_CREATION_CHANGE#	NUMBER		The creation SCN for the missing data file.
TENANT_NAME	VARCHAR2		The tenant name for this database in the Recovery Appliance.
TENANT_IDENTIFIER	NUMBER		The customer tenant identifier for this database in the Recovery Appliance.

26.32 RA_SERVER

This view describes the current settings for the Recovery Appliance.

Column	Data Type	NULL	Description
STATE	VARCHAR2(13)		The state of the Recovery Appliance: ON if the Recovery Appliance is running; OFF if it is not active.
NETWORK_CHUNKSIZE	NUMBER		The size (in MB) of network messages used by the Recovery Appliance client module to communicate with the Recovery Appliance.
SCHEDULERS	NUMBER		The number of normal schedulers currently running on the Recovery Appliance. This number excludes special purpose schedulers used for tape, replication, purge_immediate, or restore operations. Note that this column is available only with Zero Data Loss Recovery Appliance software update 12.1.1.1.8 and later.
RESOURCE_WAIT_TASK_LIMIT	VARCHAR2(48)		The limitations on task concurrency caused by resource waits.
CURRENT_RUN_TIME	INTERVAL DAY(9) TO SECOND(9)		Interval for which the Recovery Appliance has been continuously operating.

26.33 RA_STORAGE_HISTOGRAM

This view describes the storage allocation history for recent time periods.

Column	Data Type	NULL	Description
NAME	VARCHAR2(128)	NOT NULL	The name of the Recovery Appliance storage location.
SL_KEY	NUMBER	NOT NULL	The primary key for this Recovery Appliance storage location in the recovery catalog.
SLOT	NUMBER		The slot (ordered by sampling time period) in the histogram.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Column	Data Type	NULL	Description
USAGE	NUMBER		The amount of space (in GB) that was allocated during the histogram slot.

26.34 RA_STORAGE_LOCATION

This view lists defined Recovery Appliance storage locations and their allocations.

Column	Data Type	NULL	Description
NAME	VARCHAR2(128)	NOT NULL	The Recovery Appliance storage location name.
SL_KEY	NUMBER	NOT NULL	The primary key for this Recovery Appliance storage location in the recovery catalog.
DISK_GROUPS	VARCHAR2(4000)		The list of names of Oracle ASM disk groups used for storage.
MIN_ALLOC	NUMBER		The minimum amount of storage (in GB) that may be allocated.
TOTAL_SPACE	NUMBER		The maximum amount of storage (in GB) that the Recovery Appliance storage location can use for backup data.
USED_SPACE	NUMBER		The amount of space (in GB) currently used in the Recovery Appliance storage location.
FREESPACE	NUMBER		The amount of space (in GB) available for immediate use.
FREESPACE_GOAL	NUMBER		The expected free space requirement (in GB) based on usage history. Purges may occur to meet this goal.
LAST_CHECK_FILES	TIMESTAMP(6) WITH TIME ZONE		The most recent time that files were checked for consistency.
SYSTEM_PURGING_SPACE	NUMBER		The amount of space (in GB) reserved for purging operations.
PURGING_DB_KEY	NUMBER		The db_key for the database (if it exists) currently being purged in storage location.
UNRESERVED_SPACE	NUMBER		The remaining space (in GB) that is available to be assigned to database disk space reservations.
AUTOTUNE_SPACE_LIMIT	NUMBER		The maximum database reserved_space size (In GB) for which autotuning will perform unlimited reserved_space growth. NULL implies no limit.

26.35 RA_STORAGE_LOCATION_HISTORY

This view contains a daily history of storage location for the for the previous year. It has one row for each storage location for each day going back a year. It may contain phantom rows for storage locations that were deleted.

This data is not expected to be used as-is, but rather as part of GROUP BY classes in order to obtain data such as:

- Average space usage in the storage location per month over the year.
- Minimum or maximum free space goal in the previous year.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Column	Data Type	NULL	Description
SAMPLE_TIME	NUMBER		The day when this storage location history record was created.
NAME	VARCHAR2(128)	NOT NULL	The Recovery Appliance storage location name.
SL_KEY	NUMBER	NOT NULL	The primary key for this Recovery Appliance storage location in the recovery catalog.
TOTAL_SPACE	NUMBER		The maximum amount of storage (in GB) that the Recovery Appliance storage location can use for backup data.
USED_SPACE	NUMBER		The amount of space (in GB) currently used in the Recovery Appliance storage location.
FREESPACE	NUMBER		The amount of space (in GB) available for immediate use.
FREESPACE_GOAL	NUMBER		The expected free space requirement (in GB) based on usage history. Purges may occur to meet this goal.
UNRESERVED_SPACE	NUMBER		The remaining space (in GB) that is available to be assigned to the database disk space reservation.

26.36 RA_TASK

This view lists queued background tasks and their run statuses.

Column	Data Type	NULL	Description
TASK_ID	NUMBER		The ID for the task.
TASK_TYPE	VARCHAR2(30)		The type of processing performed by the task.
PRIORITY	NUMBER		The run priority for the task.
STATE	VARCHAR2(47)		The processing state for the task: EXECUTABLE, RUNNING, COMPLETED, TASK_WAIT, FAILED, and so on.
WAITING_ON	NUMBER		The ID of the task that is blocking this task when its state is TASK_WAIT.
WAITING_ON_TASK_TYPE	VARCHAR2(30)		The task_type of the WAITING_ON task.
WAITING_ON_STATE	VARCHAR2(21)		The task state of the WAITING_ON task.
CREATION_TIME	TIMESTAMP(6) WITH TIME ZONE		The time of task creation.
COMPLETION_TIME	TIMESTAMP(6) WITH TIME ZONE		The timestamp for task completion. The column is null if the task is not complete.
ELAPSED_SECONDS	NUMBER		The elapsed run time (in seconds) for the task.
ERROR_COUNT	NUMBER		Number of times that the task had errors
INTERRUPT_COUNT	NUMBER		The number of times that the task was interrupted.
LAST_INTERRUPT_TIME	TIMESTAMP(6) WITH TIME ZONE		The most recent time that the task was interrupted.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Column	Data Type	NULL	Description
EXECUTE_INSTANCE_ID	NUMBER		The ID of the database instance on which the task must run. The column is null if the task can run on any instance.
LAST_EXECUTE_TIME	TIMESTAMP(6) WITH TIME ZONE		The most recent time that the task was restarted.
DB_UNIQUE_NAME	VARCHAR2(30)		The unique name of the protected database for which the task is running.
DB_KEY	NUMBER		The primary key of the protected database for which the task is running.
SL_NAME	VARCHAR2(128)		The name of the Recovery Appliance storage location used by the task.
SL_KEY	NUMBER		The primary key of the Recovery Appliance storage location used by the task.
OSPID	VARCHAR2(128)		The platform-specific ID of the process in which the task is current running.
INSTANCE_ID	NUMBER		The ID of the database instance on which the task is currently running.
LAST_INCIDENT_ID	NUMBER		The ID of the last incident reported by the task that is currently running. Note that this column is available only with Zero Data Loss Recovery Appliance software update 12.1.1.1.8 and later.
BP_KEY	NUMBER		The key of the backup piece that is accessed by this task. Note that this column is available only with Zero Data Loss Recovery Appliance software update 12.1.1.1.8 and later.
BS_KEY	NUMBER		The key of the backup set that is accessed by this task. Note that this column is available only with Zero Data Loss Recovery Appliance software update 12.1.1.1.8 and later.
DF_KEY	NUMBER		The key of the data file that is accessed by this task. Note that this column is available only with Zero Data Loss Recovery Appliance software update 12.1.1.1.8 and later.
VB_KEY	NUMBER		The key of the virtual backup that is accessed by this task. Note that this column is available only with Zero Data Loss Recovery Appliance software update 12.1.1.1.8 and later.
HANDLE	VARCHAR2(1000)		The media manager handle that is accessed by this task. Note that this column is available only with Zero Data Loss Recovery Appliance software update 12.1.1.1.8 and later.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Column	Data Type	NULL	Description
FILENAME	VARCHAR2(4000)		The name of the backup file that is accessed by this task. Note that this column is available only with Zero Data Loss Recovery Appliance software update 12.1.1.1.8 and later.
LIB_KEY	NUMBER		The unique identifier of the SBT library that is accessed by this task. Note that this column is available only with Zero Data Loss Recovery Appliance software update 12.1.1.1.8 and later.
ARCHIVED	CHAR(1)		The archive status of the task: Y if it has moved to the archive; otherwise, N.
TENANT_NAME	VARCHAR2		The tenant name for this database in the Recovery Appliance.
TENANT_IDENTIFIER	NUMBER		The customer tenant identifier for this database in the Recovery Appliance.

26.37 RA_TIMER_TASK

Timer process tasks and their planned executions.

Column	Data Type	NULL	Description
TIMER_TYPE	VARCHAR2(54)		Purpose for running this timer task.
NEXT_EXECUTE	TIMESTAMP(6) WITH TIME ZONE		Next planned execution for this timer task.
TIMER_INTERVAL	INTERVAL DAY(9) TO SECOND(6)		Frequency for repeating this timer task.
KEY	NUMBER		The poll_key, db_key, or lib_key operated referenced by this timer task.

26.38 RA_TIME_USAGE

This view describes the Recovery Appliance elapsed and idle time for the last 30 days.

Column	Data Type	NULL	Description
TOTAL_TIME	NUMBER		The sum of the elapsed times (in seconds) across all sessions.
IDLE_TIME	NUMBER		The sum of the idle times (in seconds) across all sessions.

Copyright © 2014, 2026, Oracle and/or its affiliates.

27
rastat Utility Reference

This chapter provides details on the rastat utility. You use rastat to generate statistics to help you evaluate the performance of Recovery Appliance.

You can find the utility, `rastat.pl`, in the `/opt/oracle.RecoveryAppliance/client/` directory of a Recovery Appliance compute server.

27.1 rastat Command Syntax

```
perl rastat.pl --test=<options> --rasys=<string> --catalog=<string>
--filesize=<size>M --chunksize=<size>M --diskgroup=<string> --parms=<string>
--oracle_home=<string> --oracle_sid=<string>
```

27.2 Options

Table 27-1 rastat Options

Option	Description
`-h, --help`	Displays help information.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 27-1 (Cont.) rastat Options

Option	Description
`--test`	Specifies which of the following atomic tests to run: `[NETBACKUP|NETRESTORE|ASMREAD|ASMWRITE|CONTAINERREAD|CONTAINERWRITE|CONTAINERALLOC|ALL]` `NETBACKUP`: Measures the network performance of a protected database sending backup byte streams to the Recovery Appliance. Requires `--catalog`; `--filesize` is optional. `--parms` is also optional if it is already configured for the RMAN client. `NETRESTORE`: Measures the network performance of a protected databases receiving backup byte streams from the Recovery Appliance. Requires `--catalog`; `--filesize` is optional. `--parms` is also optional if it is already configured for the RMAN client. `ASMREAD`: Measures the disk I/O performance of the Recovery Appliance reading from an ASM disk group. Requires `--diskgroup` and `--rasys`. `--filesize` and `--chunksize` are optional. `ASMWRITE`: Measures the disk I/O performance of the Recovery Appliance writing to an ASM disk group. Requires `--diskgroup` and `--rasys`. `--filesize` and `--chunksize` are optional. `CONTAINERREAD`: Measures the disk I/O performance of the Recovery Appliance reading from container files. Requires `--diskgroup` and `--rasys`. `--filesize` and `--chunksize` are optional. `CONTAINERWRITE`: Measures the disk I/O performance of the Recovery Appliance writing to container files. Requires `--diskgroup` and `--rasys`. `--filesize` and `--chunksize` are optional. `CONTAINERALLOC`: Measures the Recovery Appliance container file allocation rate. Requires `--diskgroup` and `--rasys`. `--filesize` and `--chunksize` are optional. `ALL`: Performs all of the tests. All of the required options must be set.
`--rasys`	The connection string for the Recovery Appliance SYS account. Required for all I/O tests.
`--catalog`	The connection string for the Recovery Appliance virtual private catalog (VPC) account. Required for `NETBACKUP` and `NETRESTORE` tests.
`--filesize`	Optional. The file size in megabytes for the utility to use for the test. Setting the appropriate file size for your test requirements is highly recommended. The default file size is 1024M.
`--chunksize`	Optional. The chunk size in megabytes for the utility to use for the `CONTAINERREAD` or `CONTAINERWRITE` test. The default is the system configured chunk size.
`--diskgroup`	The name of the disk group the I/O test should read from or write to. For example, `--diskgroup=+DISK1` specifies an ASM disk group and `--diskgroup=/:DELTA` specifies a container group. Required for all I/O tests.
`--parms`	Optional if already configured in RMAN. The PARMS setting in RMAN to use for a `NETBACKUP` or `NETRESTORE` test. This parameter must specify the location of libra.so and the wallet information. For example, `--parms='SBT_LIBRARY=/u01/oracle/lib/libra.so, ENV=(RA_WALLET=location=file:/u01/oracle/dbs/ra credential_alias=ra-scan:1521/zdlra5:dedicated)'`.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Table 27-1 (Cont.) rastat Options

Option	Description
`--oracle_home`	Optional. The `$ORACLE_HOME` environment variable. Use this option to set the variable or to override the current setting.
`--oracle_sid`	Optional. The `$ORACLE_SID` environment variable. Use this option to set the variable or to override the current setting.

Copyright © 2014, 2026, Oracle and/or its affiliates.

28

Recovery Appliance Error Message Reference

This chapter provides details on the Zero Data Loss Recovery Appliance (Recovery Appliance) error messages, which occur between ranges `ORA-45100` and `ORA-45299`, between ranges `ORA-64700` and `ORA-64899`, and between ranges `ORA-61100` and `ORA-61699`.

ORA-45100: Database incarnation went from %s to %s. Recovery Appliance repair is required.
Cause: A '`startup resetlogs`' command was executed on the Recovery Appliance. This caused old metadata to be used to refer to the storage locations. Before the Recovery Appliance can be started, a repair operation must be run to synchronize its metadata with its storage.

Action: Execute `DBMS_RA.STARTUP_RECOVERY_APPLIANCE`. If any incidents are logged during the subsequent repair, they will need to be corrected. Once they have been corrected, repeat the execution of `DBMS_RA.STARTUP_RECOVERY_APPLIANCE`.

ORA-45102: unable to allocate %s bytes of storage
Cause: The Recovery Appliance was unable to allocate additional disk space in the storage location of the database for the current allocation. This condition may be due to one of the following reasons:

- Guaranteed copy has been specified for a database and there are too many backups waiting to be copied to tape.
- Nothing can be purged within the storage location of the database.
- There are no purgable backup pieces for the database and the sum of the sizes of the pieces being received by the database exceeds its `DISK_RESERVED_SPACE`.
- The metadata of the Recovery Appliance is being repaired.

Action: Add additional storage to the storage location of the database. Increase the `DISK_RESERVED_SPACE` for the database if it has been set too low.

ORA-45109: metadata for database %s; file %s has inconsistencies
Cause: Internal self checks found inconsistencies in the metadata used to manage the Recovery Appliance block pool.

Action: Contact Oracle Support Services and provide trace and alert files.

ORA-45111: Task %s is being terminated after %s restarts.
Cause: A Recovery Appliance task generated too many errors. Following an error, a task is normally restarted. If it fails to restart after 10 tries, the Recovery Appliance marks the task as broken and no longer tries to restart it.

Action: Correct the error that terminated the task and request the task to be rerun.

ORA-45113: Recovery Appliance internal error %s
Cause: An internal error was encountered.

Action: Contact Oracle Support Services and provide trace and alert files.

Copyright © 2014, 2026, Oracle and/or its affiliates.

ORA-45114: file "%s" not referenced by metadata for storage location %s
Cause: A consistency check performed by the check files task of the Recovery Appliance identified that the file was not being referenced by the metadata of the Recovery Appliance. Without these references, the Recovery Appliance cannot manage the file.

Action: If the file was inadvertently put in the storage location, it should be moved elsewhere. If the file has been separated from its metadata, contact Oracle Support Services and provide trace and alert files.

ORA-45115: database with DB_KEY %s is too big to move.
Cause: An attempt was made to move the specified database to a new storage location, but the database could not be shrunk to within its storage reservation and still preserve its retention window.

Action: Increase the storage reservation for the database or shrink its retention window.

ORA-45116: anomaly detected while reading metadata for backup piece with BP_KEY %s
Cause: A transient anomaly was found in the backup data.

Action: If the anomaly persists, find a copy of the backup piece, if available, and reinsert it into the storage location. If no copy is available, generate a new level 0 backup for all data files in the backup piece.

ORA-45117: There is not enough space for this task.
Cause: Space could not be allocated from the storage location to support this operation.

Action: Review space usage by the different databases and take action:

- Review the `RA_INCIDENT_LOG` and see what space related incidents might have been recorded. Look for failures that might have impacted normal operations.
- Look for databases without an entry in `RA_DISK_RESTORE_RANGE`. Such databases are unable to release space properly. These databases are not fully recoverable.
- Consider moving databases to another RA.
- Consider adding more space to your storage location.

ORA-45118: servlet timeout error
Cause: A restore task was waiting on a servlet process to pass data to a client. The time allotted for responding was exceeded and the restore task was aborted.

Action: This can be a common occurrence if the client cancels the restore request. Reissue the request.

ORA-45119: received a nonexistent operation for privilege change
Cause: An illegal option was specified.

Action: `BACKUP`, `RECONCILE`, `READ`, `WRITE`, and `NULL` are only supported values.

ORA-45120: operation failed due to insufficient space
Cause: The storage location was too small to support the new database.

Action: Increase the size of your storage location or reduce `DISK_RESERVED_SPACE` in the protection policy.

ORA-45121: received an incorrect value for a privilege change
Cause: An internal error was detected while granting or revoking privileges.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Action: Contact Oracle Support Services and provide trace and alert files.

ORA-45122: invalid size or number specified
Cause: An invalid size or number was specified.

Action: Use a non-NULL or number greater than 0.

ORA-45123: The name %s (%s) already exists.
Cause: The object name was not unique.

Action: Specify a unique name for this object.

ORA-45124: Object %s (%s) is referenced and cannot be deleted.
Cause: The object was in use by a storage location or database.

Action: Delete all objects that reference this item.

ORA-45125: Object %s (%s) did not exist.
Cause: The object name did not exist.

Action: Specify an existing object.

ORA-45126: failed to delete database %s
Cause: The database could not be deleted. An unexpected error has occurred.

Action: Examine the associated messages to determine the cause of the exception.

ORA-45127: Required parameter %s must be specified.
Cause: The parameter was not supplied to API routine.

Action: Rerun the command specifying the missing parameter.

ORA-45128: backup piece %s in database %s is not referenced by the catalog
Cause: A consistency check performed by the check files task of the Recovery Appliance identified that the specified backup piece was unreferenced by the RMAN catalog. Without this reference, the Recovery Appliance cannot reclaim space used by this piece.

Action: Contact Oracle Support Services.

ORA-45129: expected %s bytes used by database %s in storage location %s, but found %s bytes used
Cause: A consistency check performed by the check files task of the Recovery Appliance identified that the storage usage of the database in a storage location did not match the sum of the size of storage pieces assigned to the Recovery Appliance.

Action: Contact Oracle Support Services.

ORA-45130: Storage parameter overlaps with storage in %s.
Cause: A parameter for a storage location was specified that overlapped storage previously assigned to another storage location.

Action: Reissue the command specifying a different location for the storage.

ORA-45131: illegal or unknown restore compression option specified
Cause: The specified compression option was not supported on either the Recovery Appliance database or the database providing the backup.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Action: Query V$RMAN_COMPRESSION_ALGORITHM view to ensure the algorithm name matches one of the algorithm names in that table and that the option has IS_VALID = 'YES' and that the INITIAL_RELEASE column is less than both the Recovery Appliance and the database providing the backup. Reissue the command specifying a valid compression algorithm name.

ORA-45132: corrupt block detected in backup piece
Cause: A corrupt block was detected in a backup piece when populating the Recovery Appliance block pool.

Action: Perform block media recovery on the corrupt blocks of the database and do a cumulative level 1 backup.

ORA-45133: expected %s byte allocation by database %s, but found %s bytes allocated
Cause: A consistency check performed by the check files task of the Recovery Appliance found that the storage allocations of the database did not match the sum of the size of allocations for that database in all storage locations.

Action: Contact Oracle Support Services.

ORA-45135: request terminated by the Recovery Appliance
Cause: A request was holding resources needed by the Recovery Appliance and was terminated to free those resources. This can be the result of a lack of disk space or some other resource.

Action: Check available disk space, as well as for errors on the Recovery Appliance database.

ORA-45136: invalid value for parameter %s
Cause: The value supplied for the specified parameter was invalid.

Action: Check the Recovery Appliance documentation and rerun the command with a correct value.

ORA-45137: unknown platform
Cause: The Recovery Appliance has not received any backups from System Backup to Tape (SBT) or through polling. This is necessary for the Recovery Appliance to learn about the protected platform of the database and for the current operation to succeed.

Action: Backup a small archived log or other backup using SBT or by sending it to the polling location. Then retry this operation.

ORA-45138: Backup not found.
Cause: The specified backup could not be found in the catalog.

Action: Please check and specify the correct backup piece key or backup set key.

ORA-45139: A useful backup could not be found to correct this corruption.
Cause: A virtual backup piece key was provided, but there was no known backup on tape or disk to correct this backup.

Action: If the broken backup is the oldest virtual backup for the data file, sometimes an even older backup will have the data needed to correct the catalog. Find and specify that older backup directly.

ORA-45140: cannot insert backup into catalog
Cause: The specified backup was either not an incremental or not in the proper SCN range to correct problems in the catalog.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Action: Make sure you have provided the correct key value and find a proper incremental backup piece.

ORA-45141: File "%s" was missing from storage location %s.
Cause: During recovery of the Recovery Appliance, the specified file was referenced by the metadata of the Recovery Appliance, but was not found in its storage location.

Action: The file should be recovered from a replicated Recovery Appliance if it exists. If the file has been separated from its metadata, then contact Oracle Support Services and provide trace and alert files.

ORA-45142: The Recovery Appliance prerequisite is already set up.
Cause: The `DBMS_RA_INSTALL` procedure was executed to set up prerequisite objects for creation of a catalog schema for the Recovery Appliance. This error is reported because there can be only one user schema that manages the Recovery Appliance for the database.

Action: To re-create the Recovery Appliance schema in another user schema, uninstall the earlier Recovery Appliance schema setup.

ORA-45143: The Recovery Appliance prerequisite setup administors user name is mismatched.
Cause: The `DBMS_RA_INSTALL` procedure was executed to uninstall the Recovery Appliance prerequisite object for the wrong user name.

Action: Correct the user name parameter for `DBMS_RA_INSTALL` and reexecute the procedure.

ORA-45144: Undefined initial replication type for protection policy.
Cause: The initial_replication_type was undefined for the protection policy.

Action: Update the protection policy initial_replication_type with one of `LAST` `FULL`, `ALL`, or `NONE`.

ORA-45145: Recovery Appliance user %s does not exist.
Cause: The Recovery Appliance user did not exist.

Action: Specify an existing Recovery Appliance user.

ORA-45146: Storage location %s needs %s additional bytes of storage.
Cause: The metadata of the Recovery Appliance was being repaired following a database open with the '`resetlogs`' command and the storage allocated in the specified storage location was insufficient. This may be caused by either an '`update_storage_location`' call being lost due to the 'resetlogs' command or the storage location becoming very low on free storage when the resetlogs command was executed.

Action: Update the storage location with the specified values and try the repair again by executing `DBMS_RA.STARTUP_RECOVERY_APPLIANCE`.

ORA-45147: Database %s and database %s are both moving.
Cause: The metadata of the Recovery Appliance was being repaired following a database open with 'resetlogs' and two databases were found to be moving between storage locations. The Recovery Appliance will only function correctly when one database is being moved. This may be caused by an 'update_protection_policy' or 'update_db' call being lost due to the 'resetlogs' command.

Action: Determine the storage locations used by each database and repeat any database movements that may have been lost.

Copyright © 2014, 2026, Oracle and/or its affiliates.

ORA-45148: must fix %s errors before restarting the Recovery Appliance
Cause: During a repair of the metadata of the Recovery Appliance, errors were found that precluded the restart of the Recovery Appliance.

Action: Fix the identified errors and execute `DBMS_RA.STARTUP_RECOVERY_APPLIANCE` to retry the repair.

ORA-45149: unknown task type: %s
Cause: The Recovery Appliance tried to execute a task with an unknown task type.

Action: Contact Oracle Support Services.

ORA-45150: File %s references unknown DBID %s.
Cause: During a repair of the metadata of the Recovery Appliance, the specified file was found that referenced the specified database which was unknown to the Recovery Appliance. This may be caused by an '`add_db`' call being lost due to a '`resetlogs`' command.

Action: Repeat the lost '`add_db`' call.

ORA-45151: bad locking protocol for lock %s
Cause: An internal error caused locking to be used incorrectly.

Action: Contact Oracle Support Services.

ORA-45152: bad backup piece format for %s
Cause: During a repair of the metadata of the Recovery Appliance, the specified file was found whose type could not be determined.

Action: Remove the corrupted file and reexecute `DBMS_RA.STARTUP_RECOVERY_APPLIANCE`.

ORA-45153: unknown data file %s for DBID %s
Cause: During repair of the metadata of the Recovery Appliance, data for the specified database was found whose data file could not be found.

Action: Using RMAN on the specified database, use the '`resync`' command to refresh the metadata on the Recovery Appliance and retry the restart of the Recovery Appliance.

ORA-45154: bad Recovery Appliance format found in file %s
Cause: During a repair of the metadata of the Recovery Appliance, a file was found in a storage location that was neither a chunk file nor a backup piece.

Action: Remove the offending file from the storage location of the Recovery Appliance and retry the restart of the Recovery Appliance.

ORA-45155: The Recovery Appliance has not been installed.
Cause: The Recovery Appliance was never installed on this database. The requested procedure is only supported on the Recovery Appliance.

Action: Do not attempt the procedure except on the Recovery Appliance.

ORA-45156: SBT job %s not found
Cause: The specified SBT job was not found.

Action: Check if the SBT job has been deleted by user. If so, then drop the scheduler job.

ORA-45157: Parameter value %s (%s) is invalid.
Cause: The specified value for the parameter was invalid.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Action: Specify a valid value.

ORA-45158: SBT library %s is not ready.
Cause: The specified SBT library was found to not be ready.

Action: Check if the library has been paused by user. If so, then resume the SBT library.

ORA-45159: RECOVERY_WINDOW_GOAL is lost for database %s.
Cause: A low space condition forced the deletion of backups needed to support the recovery window goal for the named database.

Action: This is a warning and no action is needed. However, you may use `DBMS_RA.UPDATE_PROTECTION_POLICY` to increase the `DISK_RESERVED_SPACE` value of the database to ensure additional backups are saved. Select `SPACE_USAGE` from `RA_DATABASE` to see how much space is currently in use. You should also check for `KEEP` backups consuming space on disk and decide if they should be moved to tape or other disk storage. See `DBMS_RA.MOVE_BACKUP_PIECE` for more details.

ORA-45160: Incremental forever strategy is lost for database %s.
Cause: A low space condition has forced the deletion of backup data needed to generate the last remaining virtual `LEVEL 0` of one or more data files. The next client backup will be a full `LEVEL 0` backup, even if `LEVEL 1` was specified.

Action: This is a warning and no action is needed. However, you may use `DBMS_RA.UPDATE_DB` to increase the `RESERVED_SPACE` value of the database to ensure additional backups are saved. Select `SPACE_USAGE` from `RA_DATABASE` to see how much space is currently in use. You should also check for `KEEP` backups consuming space on disk and decide if they should be moved to tape or other disk storage. See `DBMS_RA.MOVE_BACKUP_PIECE` for more details.

ORA-45161: The backup piece size cannot exceed database DISK_RESERVED_SPACE.
Cause: An individual backup piece exceeded the database protection policy `DISK_RESERVE_SPACE` value. A safe `DISK_RESERVED_SPACE` value would exceed the size of the database.

Action: Use `DBMS_RA.UPDATE_DB` to increase the `DISK_RESERVED_SPACE` value of the database.

ORA-45162: System global area memory is configured incorrectly.
Cause: Check initialization parameters `LARGE_POOL_SIZE` and `SHARED_POOL_SIZE`. The Recovery Appliance will use all of `LARGE_POOL_SIZE` or 20% of `SHARED_POOL_SIZE` to restore virtual or tape backups. The actual space needed is `NETWORK_CHUNKSIZE * 2 * (number of concurrent restore channels)` where `NETWORK_CHUNKSIZE` is set using `DBMS_RA.CONFIG`. Use `DBMS_RA.CONFIG` to lower `NETWORK_CHUNKSIZE` or preferably, increase either `LARGE_POOL_SIZE` or `SHARED_POOL_SIZE`.

Action: Check initialization parameters `LARGE_POOL_SIZE` and `SHARED_POOL_SIZE` and set it correctly.

ORA-45163: operation is only supported for user %s
Cause: An attempt was made to start the Recovery Appliance by a user other than the Recovery Appliance administrator.

Action: Only start the Recovery Appliance as the user specified at installation time.

ORA-45164: The Recovery Appliance is not running.
Cause: An attempt was made to use the Recovery Appliance, but the Recovery Appliance has been deactivated by the administrator.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Action: Have the Recovery Appliance administrator execute DBMS_RA.STARTUP_RECOVERY_APPLIANCE and retry the operation.

ORA-45165: Recovery Appliance backup piece with BP_KEY %s is corrupt
Cause: Corruption was found in the backup data.

Action: Ensure that you have a functioning backup of the affected data file. Then delete the corrupt backup piece to clear the condition.

ORA-45166: unable to access file %s
Cause: An attempt was made to access the specified file which resulted in an error. An explanation of the error appears in the following messages.

Action: Verify the correctness of the file. If it is corrupt, either delete it or replace it.

ORA-45167: unable to validate backup piece with BP_KEY %s
Cause: An attempt was made to validate the specified backup piece which resulted in an error. An explanation of the error appears in the following messages.

Action: Verify the correctness of the backup piece. If it is corrupt, either delete it or replace it.

ORA-45168: unexpected executer exit while processing task ID %s of type %s
Cause: A task failed with an unexpected error code in the Recovery Appliance.

Action: Contact Oracle Support Services.

ORA-45169: unexpected timer process exit
Cause: The timer process failed with an unexpected error code in the Recovery Appliance.

Action: Contact Oracle Support Services.

ORA-45170: Storage location %s is too full.
Cause: Purging the specified storage location would result in the loss of the recovery window goal for one of its databases.

Action: Add more storage to the storage location or remove some databases from the storage location or reduce the recovery window goal for some of the databases in the storage location.

ORA-45171: The chunk optimization task has not run recently for one or more databases.
Cause: The background chunk optimization task had not been performed recently for at least one database. This may happen if the Recovery Appliance is too overloaded with foreground activities to have time to do background tasks.

Action: Remove some load from the Recovery Appliance by reducing the frequency of backups by protected databases or by offloading some of the databases from the Recovery Appliance.

ORA-45172: The validation task has not run recently for one or more databases.
Cause: The background validation task had not been performed recently for at least one database. This may happen if the Recovery Appliance is too overloaded with foreground activities to have time to do background tasks.

Action: Remove some load from the Recovery Appliance by reducing the frequency of backups by protected databases or by offloading some of the databases from the Recovery Appliance.

Copyright © 2014, 2026, Oracle and/or its affiliates.

ORA-45173: The checkfiles task has not run recently for one or more storage locations.
Cause: The background checkfiles task had not been performed recently for one or more storage locations. This may happen if the Recovery Appliance is too overloaded with foreground activities to have time to do background tasks.

Action: Remove some load from the Recovery Appliance by reducing the frequency of backups by protected databases or by offloading some of the databases from the Recovery Appliance.

ORA-45174: unable to use replication server %s
Cause: While using the Recovery Appliance, either a backup failed to be transmitted to the target replicated Recovery Appliance or a restore request failed to complete on the replicated Recovery Appliance.

Action: Check the following error messages to diagnose the actual error.

ORA-45175: unable to use SBT library %s
Cause: While using the Recovery Appliance, a request failed to complete while using the specified System Backup to Tape library.

Action: Check the following error messages to diagnose the actual error.

ORA-45176: Replication server %s is not in the paused state.
Cause: An attempt was made to update information for a replication server that was not in a paused state.

Action: Pause the replication server on this Recovery Appliance.

ORA-45177: unable to find file %s previously found while polling
Cause: A backup piece file previously found in a polling location was later not accessible to the Recovery Appliance.

Action: If the file was unavailable due to network errors, the file will be found again once the network is available. If the backup piece was deleted, create a new backup.

ORA-45178: The allocation unit size cannot be changed.
Cause: An attempt was made to move one or more databases into a storage location with a different allocation unit size. This value comes from the ASM disk group allocation unit size specified when creating the disk groups referenced by the storage location.

Action: Use a storage location with the same minimum allocation size as the source. If necessary create new disk groups with a matching size before creating a new storage location.

ORA-45179: The reconcile task has not run recently for database %s.
Cause: The background reconcile task had not been performed recently for the specified database. This may happen if the Recovery Appliance is too overloaded with foreground activities to have time to do background tasks.

Action: Remove some load from the Recovery Appliance by reducing the frequency of backups by protected databases or by offloading some of the databases from the Recovery Appliance.

Copyright © 2014, 2026, Oracle and/or its affiliates.

ORA-45180: The crosscheck task has not run recently for database one or more databases.
Cause: The background reconcile task had not been performed recently for at least one database. This may happen if the Recovery Appliance is too overloaded with foreground activities to have time to do background tasks.

Action: Remove some load from the Recovery Appliance by reducing the frequency of backups by protected databases or by offloading some of the databases from the Recovery Appliance.

ORA-45182: database access cannot be granted or revoked using catalog owner or sys
Cause: The catalog owner or `SYS` was specified as the user in the `DBMS_RA.GRANT_DB_ACCESS` and `DBMS_RA.REVOKE_DB_ACCESS` procedures. This is not allowed.

Action: A different user should be created and granted the necessary access.

ORA-45183: request is blocked by session %s on instance %s
Cause: An API request was made to the Recovery Appliance while another API was in progress. Only one API may be performed at a time.

Action: Wait for the other API to complete or kill the identified session before repeating the failed API request.

ORA-45184: ORA-%s occurred during wallet operation; WRL %s
Cause: An operation on the wallet failed due to the indicated error.

Action: Refer to the indicated Oracle message for more information.

ORA-45185: alias %s not found in %s wallet
Cause: The specified WALLET alias did not appear in the wallet.

Action: Check the WALLET alias or create an alias in the wallet for the specified attribute and retry the command.

ORA-45187: storage location %s is unusable; container repair key is %s
Cause: During a repair of the Recovery Appliance, fatal errors were detected while rebuilding the specified storage location.

Action: Inspect the alert log for the instance upon which the repair command was issued for the errors detected. If you detect that the errors are caused by missing disk groups, add those disk groups and execute the `STARTUP_RECOVERY_APPLIANCE` API. If you decide that the missing data cannot be restored, execute the `REPAIR_STORAGE_LOCATION` API with the `REPLACE` option prior to executing the `STARTUP_RECOVERY_APPLIANCE` API.

ORA-45188: storage location %s requires repair; container repair key is %s
Cause: During a repair of the Recovery Appliance, consistency errors were detected while rebuilding the specified storage location.

Action: Inspect the alert log for the instance upon which the repair command was issued for the errors detected. If you detect that the errors are caused by missing disks or disk groups, add those disks or disk groups and execute the `STARTUP_RECOVERY_APPLIANCE` API. If you decide that the missing data cannot be restored, execute the `REPAIR_STORAGE_LOCATION` API with the `REPLACE` option prior to executing the `STARTUP_RECOVERY_APPLIANCE` API.

ORA-45189: repair failed because storage location was renamed from %s to %s
Cause: During a repair of the Recovery Appliance, a storage location was found whose name was different from the name originally used to define the storage location.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Action: Delete the bad storage location and re-create it with the proper name.

ORA-45190: anomaly detected while reading metadata for database with DB_KEY %s
Cause: A transient anomaly was found in the backup data.

Action: If the anomaly persists, generate a new level 0 backup for all data files in the database.

ORA-45191: no suitable SBT library was found for the Recovery Appliance backups
Cause: Recovery Appliance could not find a suitable System Backup to Tape (SBT) library for performing the Recovery Appliance metadata backups.

Action: Create an SBT library that can be used for the Recovery Appliance backup.

ORA-45192: reservation already exists for the Recovery Appliance backup
Cause: An attempt was made to create a new reservation to perform a Recovery Appliance metadata backup when an unexpired reservation exists.

Action: Remove the existing reservation and then create a new reservation.

ORA-45193: multiple SBT libraries are present
Cause: More than one System Backup to Tape (SBT) library was found that can be used to back up the Recovery Appliance metadata.

Action: Specify a name while reserving the SBT library.

ORA-45194: Recovery Appliance metadata backup to SBT library failed
Cause: An error occurred while backing up the Recovery Appliance metadata to the System Backup to Tape (SBT) library. The error could be caused by the SBT library configuration or an internal Recovery Appliance error.

Action: Check the SBT library configuration or `RA_INCIDENT_LOG` view.

ORA-45195: reservation wait time exceeded
Cause: A timeout occurred while waiting for the system backup to tape (SBT) library reservation.

Action: Increase the wait time for the reservation by modifying the '`_drive_wait_minutes`' configuration parameter and retry the operation.

ORA-45196: failed to unreserve existing reservation
Cause: The existing System Backup to Tape (SBT) library reservation could not be unreserved.

Action: Check the '`ERROR_LOG`' table and trace files for information about the cause of this error.

ORA-45197: SBT library %s could not be found for reservation
Cause: The Recovery Appliance could not find a System Backup to Tape (SBT) library for the given name.

Action: Check the SBT library name and retry the operation.

ORA-45198: machine is not a physical Recovery Appliance
Cause: Recovery Appliance services were attempted to start on a machine that was not a physical Recovery Appliance.

Action: The Recovery Appliance services cannot be started on this system.

Copyright © 2014, 2026, Oracle and/or its affiliates.

ORA-45199: Error %s encountered when executing %s.
Cause: An error was encountered when executing PL/SQL code. This message should be accompanied by other error message(s) indicating the cause of the error.

Action: Check the accompanying errors.

ORA-45200: HTTP status code: %s
Cause: The indicated HTTP status code was received while processing a servlet request,

Action: None

ORA-45201: additional Information: %s
Cause: The indicated additional error was received while processing a servlet request.

Action: None

ORA-45202: operation failed, retry possible
Cause: A backup, restore operation failed while processing a servlet request. The operation may be retried.

Action: This message is used by the SBT client to decide whether to retry the operation.

ORA-45203: failed to %s backup piece file "%s"
Cause: An OS operation on the specified backup piece returned an error.

Action: Check additional messages.

ORA-45204: failure while deleting from sbt library %s
Cause: An attempt to delete a file from the given SBT library failed.

Action: Check additional messages for the actual cause.

ORA-45205: Not enough reserved space for this task.
Cause: The task did not have enough reserved space to proceed. Not enough storage space was allocated for backup related operations.

Action: Review incidents described in the `RA_INCIDENT_LOG` for this database. Then use `dbms_ra.update_db` to adjust `disk_reserved_space` accordingly.

ORA-45210: resource busy, retry possible
Cause: A backup or restore operation failed while processing a servlet request. The operation may be retried.

Action: This message is used by the SBT client to decide whether to retry the operation.

ORA-45211: error encountered while sending data; error code %s
Cause: An error was encountered while sending data to the client.

Action: Check additional messages.

ORA-45212: error encountered while receiving data; error code %s
Cause: An error was encountered while receiving data from client.

Action: Check additional messages.

ORA-45213: user or role '%s' does not exist
Cause: There was no user or role with the name specified.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Action: Provide a valid user name or role.

ORA-45214: cannot convert '%s' to number
Cause: An arithmetic, numeric, string, conversion, or constraint error occurred. For example, a NULL value was assigned to a variable that was declared as NOT NULL or an integer larger than 99 was assigned to a variable declared as `NUMBER(2)`.

Action: Change the data, how it is manipulated, or how it is declared so that values do not violate defined constraints.

ORA-45215: cannot delete a replication server that is in use
Cause: An attempt was made to delete a replication server that was actively restoring a backup.

Action: Cancel the restore or wait for the restore to complete before deleting the replication server.

ORA-45216: backup metadata of %s (%s) for database %s was not found
Cause: The reported backup metadata was not found.

Action: This is an informational message. Ensure that you retain the backups until SBT tasks are executed.

ORA-45217: SBT task %s is not found
Cause: The specified SBT task was not found.

Action: Provide a valid SBT task identifier and retry the command.

ORA-45264: error encountered during Recovery Appliance test recovery %s
Cause: As part of Recovery Appliance protection, test recovery was performed by the Recovery Appliance metadata protection script. The test recovery uses the image copies, without additional disk space requirement to restore datafiles, to test the database.

Action: Additional information regarding this failure is recorded in the `RA_INCIDENT_LOG` and is also displayed in the Oracle Enterprise Manager console.

ORA-45265: error encountered during Recovery Appliance backup health check %s
Cause: As part of Recovery Appliance protection, backup health check was performed by the Recovery Appliance metadata protection script. The backup health check uses the database 'validate' and 'preview' command to test the backups.

Action: Additional information regarding this failure is recorded in the `RA_INCIDENT_LOG` and is also displayed in the Oracle Enterprise Manager console.

ORA-45266: error encountered during Recovery Appliance database health check %s
Cause: As part of Recovery Appliance protection, database health check was performed by the Recovery Appliance metadata protection script. The database health check uses the 'backup validate' command to perform database health check.

Action: Additional information regarding this failure is recorded in the `RA_INCIDENT_LOG` and is also displayed in the Oracle Enterprise Manager console.

ORA-45267: Inconsistency found while validating db_key %s df_key %s
Cause: Internal self checks found inconsistencies in the metadata used to manage the Recovery Appliance block pool.

Action: Contact Oracle Support Services and provide trace and alert files.

Copyright © 2014, 2026, Oracle and/or its affiliates.

ORA-45268: Inconsistency found while purging db_key %s, df_key %s, vb_key %s
Cause: Internal self checks found inconsistencies in the metadata used to manage the Recovery Appliance block pool while purging. No blocks have been deleted and backups are valid.

Action: Contact Oracle Support Services and provide trace and alert files.

ORA-45275: container: '%s'
Cause: This message reports the name of the Recovery Appliance container involved in other messages.

Action: See associated error messages for a description of the problem.

ORA-45276: could not create container
Cause: A container creation failed. There will be other messages printed in the error stack that show more details about the problem.

Action: Investigate the entire error stack. If the problem is correctable, do so and retry creating this container.

ORA-45277: New AU size %s differs from existing AU size %s in group %s.
Cause: An attempt was made to create a new container that has a different AU size than the other containers that already exist in this container group.

Action: Specify a container with the same AU size as the other containers in the container group.

ORA-45278: Container group %s is not empty.
Cause: An attempt was made to drop the specified container group but it is not empty.

Action: Either remove the remaining objects from the group or use the FORCE option. Note that the FORCE option will irretrievably delete all files in the container group.

ORA-45279: Container group %s does not exist.
Cause: The specified container group does not exist.

Action: Specify a container group that exists.

ORA-45280: Container group %s already exists.
Cause: The specified container group already exists.

Action: Specify a container group that does not exist.

ORA-45281: Size of %s bytes exceeds maximum container size of %s bytes.
Cause: You tried to create a container greater than the maximum size.

Action: Specify a smaller size.

ORA-45282: error identifying container
Cause: An error occurred while identifying a container.

Action: There will be other messages on the error stack that show details of the problem.

ORA-45283: error writing to container
Cause: An I/O error occurred while writing to a container.

Action: There will be other messages on the error stack that show details of the problem.

Copyright © 2014, 2026, Oracle and/or its affiliates.

ORA-45284: error reading from container
Cause: An I/O error occurred while reading from a container.

Action: There will be other messages on the error stack that show details of the problem.

ORA-45285: Cannot create more than %s container groups.
Cause: An attempt to add a new container group would cause the number of container groups to exceed the system maximum.

Action: Increase the container group limit.

ORA-45286: Cannot create more than %s containers.
Cause: An attempt to add a new container would cause the number of containers to exceed the system maximum.

Action: Increase the container limit.

ORA-45287: File name %s is not valid for creation.
Cause: An attempt was made to create a contained file name in an invalid format.

Action: Correct the name and retry the operation.

ORA-45289: Cannot reserve %s bytes in container group %s.
Cause: The specified container group was out of space.

Action: Add another container to the container group.

ORA-45290: Cannot shrink file %s because file is busy.
Cause: The specified file cannot be reduced in size because some other process was holding the file open.

Action: Wait until the other process releases the file before attempting to reduce its size.

ORA-45291: Container %s is not globally identified.
Cause: An attempt to create or identify a file failed because a required container is not globally identified in this instance. The logs from the GEN0 process will usually indicate the reason why the file could not be identified.

Action: Examine the logs from the GEN0 process and correct the problem that is making some containers inaccessible.

ORA-45292: error during container group rebuild
Cause: An unrecoverable error occurred during container group rebuild.

Action: Review other messages on the error stack for additional details.

ORA-45293: Cannot shrink file
Cause: An attempt to shrink file was requested but this has been prevented because the file has been marked as not shrinkable.

Action: Nothing. File shrink is aborted.

Next grouping of error messages.

ORA-57800: Blocks metadata storage internal I/O error %s.
Cause: The Recovery Appliance encountered an internal error while performing a read or write operation on backup blocks metadata stored in the delta store.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Action: Contact Oracle Support Services. Provide trace files, alert files, Recovery Appliance System Activity Report, and the Datapump dump.

ORA-57801: Excess fragmentation detected while reading blocks metadata for datafile %s.
Cause: Backup blocks metadata stored in the delta store was found to be excessively fragmented for the datafile.

Action: Contact Oracle Support Services. Provide trace files, alert files, Recovery Appliance System Activity Report, and the Datapump dump.

ORA-57802:
Cause:

Action:

28.1 Recovery Appliance Error Message Reference (Continued)

This chapter provides details on the Zero Data Loss Recovery Appliance (Recovery Appliance) error messages, which occur between ranges `ORA-57800` and `57999`, between ranges `ORA-64700` and `ORA-64899` and between ranges `ORA-61100` and `ORA-61699`.

ORA-57800: Blocks metadata storage internal I/O error %s.
Cause: The Recovery Appliance encountered an internal error while performing a read or write operation on backup blocks metadata stored in the delta store.

Action: Contact Oracle Support Services. Provide trace files, alert files, Recovery Appliance System Activity Report, and the Datapump dump.

ORA-57801: Excess fragmentation detected while reading blocks metadata for d atafile %s.
Cause: Backup blocks metadata stored in the delta store was found to be excessively fragmented for the datafile.

Action: Contact Oracle Support Services. Provide trace files, alert files, Recovery Appliance System Activity Report, and the Datapump dump.

ORA-57803: Minimum recovery needed for database %s is more than the specified threshold of %s days.
Cause: One or more datafiles haven't been backed up recently leading to the `minimum_recovery_needed` value exceeding the threshold.

Action: Query the `RA_MIN_REC_NEEDED_DIAG` view to determine which datafiles haven't been backed up recently and back them up.

ORA-57804: Backups are being held longer than `max_retention_window` for database %s.
Cause: Datafile or archivelog backups haven't been purged and are being held longer than `max_retention_window`.

Action: Query the `RA_RECOVERY_WINDOW_GOAL_DIAG` view to determine which datafiles haven't been purged and are being held longer than `max_retention_window`.

ORA-57805: Unable to delete database %s because tape/cloud backups could not be purged..
Cause: The Recovry Appliance could not delete the database because the tape/cloud backups for the corresponding database could not be purged.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Action: Ensure that the SBT library is running.

ORA-57806: MAXPIECESIZE backups are not supported in the block pool. BP_KEY %s.
Cause: Incremental backups with MAXPIECESIZE are not supported in the block pool.

Action: Remove the MAXPIECESIZE clause from the CONFIGURE or ALLOCATE channel command.

ORA-57807: The Recovery Appliance export backup bundles do not get created: %s
Cause: The Oracle user password expired on one or more compute nodes.

Action: Verify the status of the Oracle user account by running "`racli run check --check_name oracle_password`". Update the account setting if necessary. Run the export process manually to verify error-free operation as described in the Owner's Guide.

ORA-57808: Failed to open file %s.
Cause: The Recovery Appliance could not open the specified file because the path is incorrect, the file is missing, or operating system permissions prevent access.

Action: There was a problem opening the specified file. Contact Oracle Support Services.

ORA-57809: Downstream Recovery Appliance does not support %s.
Cause: The downstream Recovery Appliance is running older software that does not support the requested feature.

Action: Upgrade the software on the downstream Recover Appliance.

ORA-64700: Recovery Appliance is shutting down
Cause: The Recovery Appliance was in the process of shutting down. This message is recorded in the incident log for the Recovery Appliance. When the shutdown completes, the incident is marked as `FIXED`.

Action: Wait for the Recovery Appliance to complete its shutdown.

ORA-64701: Storage location %s can no longer honor its reservations.
Cause: The specified storage location did not contain enough space to fulfill the reservations of all of the databases assigned to it. This error will be seen if a storage location lost part of its disk space and was in the process of being repaired.

Action: Either shrink the reservations for the databases contained within the storage location or add additional disk space to the storage location.

ORA-64702: Error repairing container files for storage location %s: %s
Cause: An error was returned while trying to rebuild or repair the container files used to store data from protected databases.

Action: The subsequent error will identify the error that needs to be addressed before the Recovery Appliance can be repaired.

ORA-64703: resource error detected
Cause: A task needed to be interrupted because it detected a resource limitation such as insufficient temporary table space or a snapshot being too old. It will be retried once the contention for the resource decreases. The secondary messages will identify the resource that has been exhausted.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Action: If this error occurs rarely, no user action is required. If the condition becomes persistent, the Recovery Appliance administrator should increase the resource that is exhausted.

ORA-64705: No destination in "%s" at column %s
Cause: The storage destination contained a syntax error.

Action: Correct the syntax error and retry the operation.

ORA-64708: more than one polling_location in "%s" at column %s
Cause: More than one `polling_location` was specified. Only one `polling_location` is allowed.

Action: Specify only one polling destination directory and retry the operation.

ORA-64709: ASM polling_location is not supported in "%s" at column %s
Cause: The `polling_location` specified an ASM-based location. Only non-ASM-based `polling_locations` are supported.

Action: Specify one non-ASM polling destination and retry the operation.

ORA-64711: Storage destination do not reference an ASM diskgroup.
Cause: The storage destination referenced a non-ASM storage location. Only ASM-based storage destinations are allowed. The operation has been rolled back.

Action: Correct the syntax error and retry the operation.

ORA-64713: Requested size %s for %s was too small; already using %s.
Cause: The size requested for the storage destination was smaller than its current size.

Action: Increase the requested size and retry the operation.

ORA-64714: Requested size %s for %s was larger than total available space %s.
Cause: The size requested for the storage destination was larger than its current used space plus its current free space.

Action: Decrease the requested size and retry the operation.

ORA-64715: Instance %s is not available to Recovery Appliance.
Cause: The Recovery Appliance is not utilizing the specified instance due to its absence from the Oracle RAC.

Action: Restart the specified instance or repair any connectivity issues with the specified instance.

ORA-64716: Storage location %s allocation size %s does not equal diskgroup %s allocation size %s.
Cause: The minimum allocation size of the specified storage location was not the same as the specified diskgroup allocation unit size.

Action: Specify a different diskgroup or a different storage location and retry the operation.

ORA-64717: Network chunk size %s is not a multiple of diskgroup %s allocation size %s.
Cause: The configured network chunk size was not a multiple of the specified diskgroup allocation unit size.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Action: Specify a different diskgroup or reconfigure the network chunk size and retry the operation.

ORA-64718: Diskgroup %s allocation size %s is not a power of two.
Cause: The diskgroup allocation unit size was not a power of two.

Action: Specify a different diskgroup and retry the operation.

ORA-64719: Diskgroup %s allocation size %s is less than two megabytes %s.
Cause: The diskgroup allocation unit size was less than two megabytes.

Action: Specify a different diskgroup and retry the operation.

ORA-64720: No containers were created for storage location %s.
Cause: No storage was allocated and initialized for the specified storage location.

Action: Specify a different diskgroup for the storage location, increase the size of the diskgroup, or reduce the size of the storage location and retry the operation.

ORA-64721: Reserved space %s is less than the minimum reservation %s.
Cause: No storage was allocated and initialized for the specified storage location.

Action: Specify a different diskgroup for the storage location, increase the size of the diskgroup, or reduce the size of the storage location and retry the operation.

ORA-64722: Number of drives must be greater than zero.
Cause: The specified number of tape drives was NULL or was less than or equal to zero.

Action: Specify a number of tape drives greater than 0 and retry the operation.

ORA-64723: Number of drives reserved for restore operations must be greater than or equal to zero.
Cause: The number of tape drives reserved for restore operations was NULL or was less than zero.

Action: Specify a number of tape drives reserved for restore operations greater than or equal to zero and retry the operation.

ORA-64724: Number of restore drives %s too large; must be less than %s.
Cause: The number of tape drives reserved for restore operations was at least as large as the total number of drives available. The number of tape drives reserved for restore operations must be at least one less than the total number of drives available.

Action: Specify a number of tape drives reserved for restore operations less than the total number of drives available and retry the operation.

ORA-64725: Number of streams must be greater than zero.
Cause: The number of streams was less than or equal to zero.

Action: Specify a number of streams greater than zero and retry the operation.

ORA-64726: Number of streams %s too large; must be no larger than %s.
Cause: The number of streams was larger than the total number of drives available. The number of streams must be no larger than the total number of drives available.

Action: Reduce the number of available streams and retry the operation.

Copyright © 2014, 2026, Oracle and/or its affiliates.

ORA-64727: Number of copies %s not in the range 1 through 4.
Cause: The number of copies was either NULL or not in the range 1 through 4 inclusive.

Action: Specify a number of copies in the range 1 through 4 and retry the operation.

ORA-64728: Replication server name length %s is too long.
Cause: The replication server name was longer than 128 characters.

Action: Specify a replication server name shorter than 128 characters and retry the operation.

ORA-64729: Replication server proxy port %s must be greater than zero.
Cause: The replication server proxy port number was less than or equal to zero.

Action: Specify a replication server proxy port number greater than zero and retry the operation.

ORA-64730: Replication server proxy URL provided but proxy port is NULL.
Cause: A replication server proxy URL was provided but a proxy port number was not. If either a proxy URL or a proxy port are specified, both must be specified.

Action: Specify both a replication server URL and a replication server proxy port number and retry the operation.

ORA-64731: Replication server proxy port provided but proxy URL is NULL.
Cause: A replication server proxy port number was provided but a proxy URL was not. If either a proxy URL or a proxy port are specified, both must be specified.

Action: Specify both a replication server URL and a replication server proxy port number and retry the operation.

ORA-64732: HTTP server not configured at replication host.
Cause: The HTTP server at the replication host site has not been configured.

Action: Configure the HTTP server at the replication host site and retry the operation.

ORA-64733: Unable to move individual backup piece with BP_KEY %s; not a KEEP backup.
Cause: An attempt was made to move an individual backup piece, but the backup set of which this backup piece was a member was not a KEEP backup.

Action: Specify a backup piece key that is a member of a KEEP backup set and retry the operation.

ORA-64735: Unknown incarnation detected at Recovery Appliance, need catalog resync.
Cause: A new archived log or backup set belonging to the new incarnation was received at Recovery Appliance.

Action: Using RMAN, connect to the Recovery Appliance as a recovery catalog, primary database as target database and perform the resynchronization operation using the `RESYNC CATALOG` RMAN command. If this error occurred at downstream of the Recovery Appliance (in a replicated Recovery Appliance setup), the reconcile operation fixes this error automatically when the same error is fixed at the upstream Recovery Appliance.

Copyright © 2014, 2026, Oracle and/or its affiliates.

ORA-64736: Task ID %s of type %s has been interrupted %s times.
Cause: The specified task was restarted an unexpected number of times. Tasks get interrupted when there is competition for resources. This is only a warning. It does not necessarily indicate a problem with the Recovery Appliance.

Action: If these problems persist for long periods, contact Oracle Support Services.

ORA-64737: Unable to copy a full backup for database %s because of missing data files.
Cause: While creating a full database backup to tape or to a replicated Recovery Appliance, level 0 backups of one or more data files were missing.

Action: Query the `RA_SBT_TEMPLATE_MDF` view to determine the data files for which backups are missing. If using the "incremental forever" backup strategy, perform a level-0 incremental backup for the given database and retry the operation.

ORA-64738: guaranteed copy suspended for database %s
Cause: One of the following operations was performed resulting in the database using more than its allotted disk space:
- `UPDATE_DB` lowering the `DISK_SPACE_RESERVE` value.
- `UPDATE_PROTECTION_POLICY` setting the `guaranteed_copy` parameter to `YES`
- `DELETE_SBT_LIBRARY` where backup data for the given database existed. New backup requests may be stalled until the system can recompute the safety of allowing additional backup data or backup data may be lost.

Action: This event can be avoided by ensuring backups are written to tape in a timely manner. Conversely, one should avoid the activities listed in the Cause statement when backups are not being written to tape in a timely manner.

ORA-64739: Tag length %s is too long
Cause: The tag name was longer than 32 characters.

Action: Specify a tag name shorter than 32 characters and retry the operation.

ORA-64740: Redo and backups from database %s have not been seen for more than UNPROTECTED_WINDOW period
Cause: An `UNPROTECTED_WINDOW` parameter in protection policy has been specified and the Recovery Appliance has not received sufficient on-line redo, archive log backups, or data file backups from the given target database for at least that period.

Action: Ensure that backups are being performed in a timely manner and that, if set up, redo logs or backups are being sent to the Recovery Appliance.

ORA-64741: Scheduler %s running task %s of type %s did not stop after %s requests.
Cause: The specified Recovery Appliance scheduler process could not be stopped.

Action: If these problems persist for long periods, contact Oracle Support Services.

ORA-64742: database (%s) is in state %s which is unsupported for (%s).
Cause: An operation was attempted on a database that was in a state precluding the operation being performed.

- The database is being deleted, is suspending, has been suspended or is provisional, and therefore cannot be used to allocate Recovery Appliance storage location.
- Delete database cannot be performed on a database that is already being deleted or suspended.

Copyright © 2014, 2026, Oracle and/or its affiliates.

- The database is unregistered for archival backups.
- The database is in an inactive state that cannot be used for data guard.

Action: The operation should not be tried on databases in an incompatible database state.

ORA-64744: Argument %s is null, invalid, or out of range.
Cause: The argument was expecting a non-null, valid value but the argument value passed in was null, invalid, or out of range.

Action: Check your program and correct the caller of the routine to not pass a null, invalid or out-of-range argument value.

ORA-64745: Name length is %s characters; maximum length is %s characters.
Cause: The length of the name exceeded the limit.

Action: Specify a shorter name and retry the operation.

ORA-64746: Name contains invalid characters.
Cause: The name incorrectly started with "_", "-", ":" or digits or contained non-alphanumeric characters. Verify that all other double quotation marks, if any, in the string are adjacent pairs of double quotation marks. Double quotation marks must not be used in the middle of the name.

Action: Change the name and exclude the invalid characters.

ORA-64747: Name contains invalid character "%s" at the position [%s].
Cause: The name incorrectly started with "_", "-", ":" or digits or contained non-alphanumeric characters. Verify that all other double quotation marks, if any, in the string are adjacent pairs of double quotation marks. Double quotation marks must not be used in the middle of the name.

Action: Change the name and exclude the invalid characters.

ORA-64748: trace file writing initiated using %s
Cause: The configuration of the Recovery Appliance was modified to enable the producing of trace files. Trace files have the capacity to exhaust disk space on the Recovery Appliance.

Action: Turn off the tracing when it is no longer required.

ORA-64750: Instance %s is unable to access %s.
Cause: The Recovery Appliance was unable to find a file that is required for its operation.

Action: Ensure that the file system of the specified file is available on the specified instance.

ORA-64751: Replication setup error during %s. replication server:%s database:%s.
Cause: The Recovery Appliance was unable to complete the configuration and setup of replication for the database specified.

Action: Validate that the downstream replication server is properly configured and all network communication paths are valid.

ORA-64752: storage unavailable for new redo or backups for database %s
Cause: There was a failure while backing up redo or copying backups from a polling location. This condition may be due to one of the following reasons:
* An individual backup piece exceeded the database protection policy `DISK_RESERVE_SPACE` value. * Guaranteed_copy is enabled but not enough data has been spooled to tape. * Misconfiguration of the storage location size.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Action: Check for the value of DISK_RESERVED_SPACE and storage location uses.

ORA-64753: Incorrect object type specified; specified %s, expected %s
Cause: An incorrect object type was given to an API command.

Action: Use the object-specific API. For example, use 'resume_replication_server' instead of 'resume_sbt_library'.

ORA-64754: Unable to perform operation with associated tape or replication objects.
Cause: An attempt to execute 'update_db' or 'update_protection_policy' and change storage locations with a replication server or tape job associated with the protection policy failed.

Action: Create a temporary protection policy that has the same storage location as the current protection policy with the tape and replication attributes of the target protection policy, Update to the temporary protection policy then finally update to the target protection policy.

ORA-64755: Operation is not allowed when the Recovery Appliance is not running.
Cause: An API request was attempted that requires the Recovery Appliance to be operational.

Action: Have the Recovery Appliance administrator execute DBMS_RA.STARTUP_RECOVERY_APPLIANCE and retry the request.

ORA-64757: unable to restore backup piece with BP_KEY %s
Cause: An attempt was made to restore the specified backup piece which resulted in an error. An explanation of the error appears in the following messages.

Action: Verify the correctness of the backup piece. If it is corrupt, either delete it or replace it.

ORA-64758: unable to grow delta store metadata in tablespace %s
Cause: Additional extents could not be allocated for the tables used to implement the delta store.

Action: Add additional storage to the indicated tablespace.

ORA-64759: Recovery Appliance is leaving restricted resources state
Cause: The Recovery Appliance ended its restrictions on task execution. The restricted resources state was entered when tasks could not run due to insufficient temporary table space or insufficient undo space. At that time, resource intensive tasks were put into RESOURCE_WAIT state.

Action: None. This is only an informational message entered in the alert log.

ORA-64760: Database %s has had tasks in ordering wait state for over %s days.
Cause: The specified database had an INDEX_BACKUP task that could not be run because the task did not tile against the delta store. An incremental backup piece will not tile into the delta store when the necessary control file information is missing or when the backup that it depends upon is not found in the delta store.

Action: Resolve the warnings for the database. A new full backup for the database should also clear this condition.

ORA-64761: disk group %s is not usable by the Recovery Appliance
Cause: A disk group was supplied to either the create_storage_location or update_storage_location APIs that was not previously prepared by the installation software for the Recovery Appliance.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Action: Run the `ra_update` procedure to process the disk group and retry the API.

ORA-64762: Task %s of type %s has been running for %s.
Cause: The specified task did not complete its execution after a reasonable period. This is only a warning message.

Action: If these errors persist, contact Oracle Support Services.

ORA-64763: Task %s of type %s was terminated after running for %s.
Cause: The specified task did not complete and was presumed hung. Its process was stopped and restarted.

Action: If these errors persist, contact Oracle Support Services.

ORA-64764: Metadata validation task %s was interrupted by COALESCE task while validating df_key %s.
Cause: . The metadata validation task took too long to validate a df_key and blocked a COALESCE task. The df_key validation was terminated before completion and the df_key was skipped.

Action: None

ORA-64766: backup deletion using RMAN prevented by protection policy
Cause: Recovery Manager was prevented from deleting a backup piece, because the `allow_backup_deletion` parameter of the applicable Recovery Appliance protection policy was `NO`, or because a prior `recovery_window_compliance` duration has not passed.

Action: Modify the `allow_backup_deletion` parameter of the applicable protection policy to `YES` to allow for deletion of backups and wait for any remaining `.recovery_window_compliance` windows to espire. Check the `ra_recovery_compliance` view to find active compliance windows.

ORA-64767: restore timed out
Cause: The Recovery Appliance terminated the restore operation due to unresponsiveness of the client database.

Action: Verify the network between the client database and the Recovery Appliance. Also verify the client database I/O performance. If the error persists, contact Oracle Support Services.

ORA-64768: KEEP file size %s cannot exceed available DISK_RESERVED_SPACE %s
Cause: There was no more space for a `KEEP` backup piece. The total `DISK_RESERVED_SPACE` minus the space currently consumed by `KEEP` backups was less than the size of the current `KEEP` backup piece. The piece that caused the error was deleted.

Action: Use DBMS_RA.UPDATE_DB to increase the DISK_RESERVED_SPACE value of the database or move KEEP (archival) backups to other storage using the DBMS_RA.MOVE_BACKUP function.

ORA-64771: reconcile error during %s; replication server: %s; database: %s
Cause: The specified database was unable to reconcile with the specified replication server. This is typically due to either a network communication failure, missing or incorrect credentials, incorrect wallet credentials, missing or incorrectly ordered `add_db` and `grant_db_access` calls.

Action: Check the incident log for active failures with a component type of `REPLICATION_RECONCILE`. Resolve the reconcile issues and `resume_replication_server` or

Copyright © 2014, 2026, Oracle and/or its affiliates.

wait until the next reconcile time period. After a successful reconcile all stalled replication tasks will continue and the incident is marked as FIXED.

ORA-64772: downstream replication server: %s is not accessible
Cause: The specified downstream replication server was not running, was not accesible, had incorrect credentials for the replication user in the upstream wallet, or was in the process of starting up or shutting down.

Action: If the credentials are incorrect, correct the credentials. Otherwise, the incident is marked as FIXED once the downstream replication server comes back online or is accessible over the network.

ORA-64773: Note: %s generated during execution of API command: %s
Cause: The specified command was entered correctly but an unexpected event occurred during its processing.

Action: Check the message for further information.

ORA-64774: database %s has replication tasks in reconcile wait state for over %s days
Cause: The specified database had a BACKUP_SBT task that could not be run because the task was waiting for a successful reconcile.

Action: Check the incident log for active failures with a component type of REPLICATION_RECONCILE. Resolve the reconcile issues and execute a manual reconcile, execute a resume_replication_server, or wait until the next reconcile time period. After a successful reconcile, all stalled replication tasks continue and the incident is marked as FIXED.

ORA-64775: unable to authenticate with the downstream recovery appliance for the database %s.
Cause: The specified database could not execute the required code on the downstream replication server. This is typically due to a missing add_db or grant_db_access on the downstream or the commands were issued in the incorrect order.

Action: Ensure that you execute add_db followed by grant_db_access on the downstream replication server. Then reissue the command, execute a manual reconcile, execute a resume_replication_server, or wait until the next reconcile time period. After a successful reconcile, all stalled replication tasks continue and this error is cleared.

ORA-64776: message from downstream replication server during %s: %s
Cause: An unexpected event occurred at the downstream replication server.

Action: The message includes additional information for further examination on the downstream replication server.

ORA-64777: deletion of polled file failed
Cause: The Recovery Appliance could not delete the file that was backed up from the polling location.

Action: Check if the directory specified in the polling location grants the necessary permission to allow for deletion of files.

ORA-64778: privilege not granted to %s for database: %s
Cause: The privilege that was revoked was not granted to the user for this protected database, the revoke was not required.

Action: Do not issue the revoke for this privilege for the protected database, it is not granted to the user.

Copyright © 2014, 2026, Oracle and/or its affiliates.

ORA-64779: The catalog is missing necessary control file information for database %s.
Cause: A backup arrived before the Recovery Appliance could receive all the necessary control file information. INDEX_BACKUP tasks for these data files remain in ORDERING_WAIT state until the catalog receives all the information.

Action: Use the RMAN command RESYNC CATALOG to fix this issue. If the problem persists, contact Oracle Support Services.

ORA-64780: Section 1 of multi-section backup set %s of data file %s has not reached this Recovery Appliance.
Cause: Section 1 of the multi-section backup did not reach the Recovery Appliance. Without the section, INDEX_BACKUP tasks of other sections of the backup set remain in ORDERING_WAIT state.

Action: This could be temporary because sections of a multi-section backup do not reach the Recovery Appliance in order. If the problem persists, check the status of the incoming backup, as well as status of INDEX_BACKUP tasks for this database.

ORA-64781: Section %s of the prior multi-section backup set %s of data file %s is missing.
Cause: A section of the previous multi-section backup prior to the current backup was missing. Without the section, INDEX_BACKUP tasks for processing the current backup set remain in ORDERING_WAIT state.

Action: An INCREMENTAL LEVEL 1 CUMULATIVE or INCREMENTAL LEVEL 0 backup of the data file should clear this condition.

ORA-64782: Missing backups between SCNs %s and %s for data file %s.
Cause: There was a gap in backups provided to the Recovery Appliance. Without the backup, the current backup could not be processed, and the INDEX_BACKUP task remained in ORDERING_WAIT state.

Action: An INCREMENTAL LEVEL 1 CUMULATIVE or INCREMENTAL LEVEL 0 backup of the data file should clear this condition.

ORA-64783: The backup of data file %s is tiling with a backup written to different media.
Cause: The backup on which the current backup piece depends was written to different media and no longer resides on this Recovery Appliance.

Action: The incremental forever mechanism of the Recovery Appliance does not support backups written to disk or other media. Provide an INCREMENTAL LEVEL 0 backup of the data file.

ORA-64784: Corruption found in backup %s of data file %s in Recovery Appliance. New backup cannot be processed.
Cause: A new backup could not be processed because the most recent backup piece of the data file residing in the Recovery Appliance was corrupt.

Action: An INCREMENTAL LEVEL 0 backup of the data file should clear this condition.

ORA-64785: Online redo shipping for DBID %s cannot be performed because the Recovery Appliance is not available.
Cause: The Recovery Appliance was unable to automatically receive archive logs from the indicated database because the Recovery Appliance was not running.

Action: Execute the startup API on the Recovery Appliance before sending archive logs to it.

Copyright © 2014, 2026, Oracle and/or its affiliates.

ORA-64786: Online redo shipping for DBID %s cannot be performed because the delta store is under repair.
Cause: The Recovery Appliance was unable to automatically receive archive logs from the indicated database because the delta store was under repair.

Action: The archive logs are accepted once the repair completes. If the message persists, contact Oracle Support Services.

ORA-64787: Recovery Window Goal is too small.
Cause: The protection policy is having the Recovery Window Goal is set to zero (0) which is not allowed for the Recovery Appliance.

Action: Please update the protection policy with a non zero or a NULL Recovery Window Goal.

ORA-64788: Online redo shipping for DBID %s cannot be performed because its platform is unknown.
Cause: The Recovery Appliance was unable to automatically receive archive logs from the indicated database because an initial backup from the database was not supplied. The initial backup provides the platform identification to the Recovery Appliance.

Action: An initial backup must be sent from the indicated database to the Recovery Appliance using the recovery manager (RMAN).

ORA-64789: Online redo shipping for DBID %s cannot be performed because its storage location needs repair.
Cause: The database for the Recovery Appliance was opened by specifying resetlogs.

Action: Contact Oracle Support Services.

ORA-64790: Online redo shipping for DBID %s cannot be performed because it is not known by the Recovery Appliance.
Cause: The database was not identified in the catalog of the Recovery Appliance.

Action: Execute the `ADD_DB` API for the specified database.

ORA-64791: DBMS_SCHEDULER jobs have delayed starting on instance %s.
Cause: Problems on the specified instance are preventing `DBMS_SCHEDULER` jobs from getting into `RUNNING` state. Until the backlog is cleared, only jobs essential to the health of the Recovery Appliance are created on the instance.

Action: Contact Oracle Support Services.

ORA-64792: incorrect catalog_user_name parameter specified in create_replication_server
Cause: An incorrect `catalog_user_name` parameter was specified in the `create_replication_server` or `update_replication_server` API.

Action: The catalog_user_name parameter must be the owner of the Recovery Appliance schema.

ORA-64793: unexpected scheduler exit
Cause: The task scheduler failed with an unexpected error code in the Recovery Appliance.

Action: Contact Oracle Support Services.

Copyright © 2014, 2026, Oracle and/or its affiliates.

ORA-64794: unable to access the encryption HSM wallet or the encryption wallet is not open on the current instance
Cause: Encryption required a Hardware Security Module (HSM) wallet that was open across all instances.

Action: Ensure the following: - That there is a valid connection to the HSM keystore. - That the HSM credentials are correct. - That the wallet is open across all Oracle RAC instances in `GV$ENCRYPTION_WALLET`.

ORA-64795: unable to access or create the encryption key
Cause: Encryption could not create or obtain the required access to a valid backup encryption key during the copy to tape operation.

Action: This could be due to a network connection issue, missing or incorrect credentials, or a misconfigured key store. Inspect the `ERROR_TEXT` in the `RA_INCIDENT_LOG` for additional information.

ORA-64796: unable to create encrypted backup during copy to tape
Cause: During the copy to tape operation, backup encryption could not create or obtain the required access to a valid encryption key.

Action: Check the `ERROR_TEXT` in the `RA_INCIDENT_LOG` for additional information.

ORA-64797: waiting for recall of backup piece %s from cloud storage
Cause: Restore was waiting for a recall of the backup piece from cloud storage.

Action: None. This is only an informational message. The incident status is changed to `FIXED` once the backup piece is recalled.

ORA-64798: encryption key does not exist in wallet of executing instance
Cause: The protected databases current encryption key did not exist in the wallet of the Oracle RAC instance or the instance that was executing the copy to tape operation.

Action: Rekey the current database and verify that the `RA_DATABASE.ENCRYPTION_KEYID` for the protected database exists across all instances in `GV$ENCRYPTION_KEYS.KEY_ID`.

ORA-64799: Job template requires encryption for the specified library.
Cause: Encryption was not specified for a library that requires encryption.

Action: Use the `DBMS_RA.UPDATE_SBT_JOB_TEMPLATE` procedure and specify an encryption algorithm for all job templates associated with the specified library.

ORA-64800: unable to create encrypted backup for big endian platform
Cause: Copy to tape operation could not create encrypted backup for big endian platform.

Action: Contact Oracle Support Services.

ORA-64801: Altering RASYS password in SQL is not allowed.
Cause: RASYS Password modification attempted via SQL.

Action: Use `racli alter rasys_user --password`.

ORA-64803: failure to get ingest name for archivelog grouping
Cause: The Oracle backup process was unable to get the ingest name needed to connect to the Oracle restore process.

Action: Set the ingest name for the local node in the configuration variable.

Copyright © 2014, 2026, Oracle and/or its affiliates.

ORA-64804: cannot reinsert deleted backup into delta store
Cause: The specified backup could not be reinserted into the delta store, because it either was previously deleted or contained a corresponding virtual backup in the same backup set that was removed from the catalog.

Action: Take a new backup of the data file.

ORA-64805: cannot reinsert virtual full backup into delta store
Cause: Virtual full backups or copies of virtual full backups could not be reinserted into the delta store because it could corrupt existing backups of the database.

Action: Use a virtual incremental backup of the data file.

ORA-64806: incorrect or all NULL input parameters specified for reset_error API
Cause: Some or all of the input parameters specified for `reset_error` API were incorrect or they were all NULL input parameters.

Action: Provide at least one valid input parameter.

ORA-64807: Datafile encryption information for database %s, data file %s does not match.
Cause: A data file was encrypted or rekeyed on the protected database and now the encryption key for the current `INCREMENTAL LEVEL 1` backup does not match the prior encryption key.

Action: An explicit `INCREMENTAL LEVEL 0` backup is needed for the encrypted data files to maintain recoverability.

ORA-64808: Initialization parameter %s has more than one SID value specified
Cause: There were multiple instance settings for one initialization parameter.

Action: Use "`racli update parameter -type=db`" to update the initialization parameter.

ORA-64809: Initialization parameter %s does not match the recommended value
Cause: Initialization parameter did not match the recommended value.

Action: Use "`racli update parameter -type=db`" to update the initialization parameter.

ORA-64810: Initialization parameter %s is not set with the proper value
Cause: Initialization parameter was not set with the proper value.

Action: Set the proper initialization parameter value.

ORA-64811: Initialization parameter %s has a non-default value.
Cause: Initialization parameter had a non-default value.

Action: Use "`racli update parameter -type=db -restart_db`" to update the initialization parameter.

ORA-64812: -force option is required to ignore checking %s initialization parameter
Cause: An attempt was made to ignore checking the initialization parameter value without specifying the -force option.

Action: Add the -force option when attempting to ignore checking the initialization parameter.

Copyright © 2014, 2026, Oracle and/or its affiliates.

ORA-64813: Task %s is being suspended.
Cause: A Recovery Appliance task generated too many errors. Following an unexpected error, a task will be restarted. If it fails to successfully run after 10 tries, the Recovery Appliance normally marks the task as broken and no longer tries to restart it. However, if a task is important for a database's recoverability, it will not be failed. Instead, the task will be suspended. Suspended tasks will be retried every few days until they successfully run or have an irrecoverable failure.

Action: Contact Oracle Support Services if the task remains in SUSPENDED state for more than two weeks.

ORA-64814: Downstream Recovery Appliance does not support remoteRA messages.
Cause: The Recovery Appliance logged this diagnostic message for a remoteRA operation in which the downstream was an older version.

Action: Consider upgrading the downstream Recovery Appliance software to match the upstream.

ORA-64816: Cannot create a replication server to an old version downstream Recovery Appliance.
Cause: The downstream Recovery Appliance was an older version.

Action: Contact Oracle Support Services.

ORA-64817: Error interacting with downstream Recovery Appliance during %s; replication server: %s; database: %s
Cause: An error was encountered during communication or conversationwith a downstream Recovery Appliance for the specified database with the specified replication server. This is typically due toa network communication failure, missing or incorrect wallet,credentials, missing or incorrectly ordered `add_db` and `grant_db_access` calls. The error could also indicate an issue during the conversation with the remote Recovery Appliance.

Action: Check the incident log for active failures with a component type of `REPLICATION_REMOTERA`. Resolve the connection issues and execute the `resume_replication_server` command or wait until the next reconcile time period. After a successful reconcile, all stalled replication tasks continue and the incident is marked as `FIXED`.

ORA-64818: Polling feature not supported for multiple tenants
Cause: Polling feature did not work on multiple tenants.

Action: None.

ORA-64819: Database %s with specified %s (%s) does not exist.
Cause: The specified database did not exist in the specified tenant.

Action:Specify an existing database and its associated tenancy.

ORA-64820: db_key %s does not exist.
Cause: No database associated with the `db_key` specified was found in the Recovery Appliance.

Action: Specify an existing `db_key`.

ORA-64821: df_key %s does not exist.
Cause: No data file associated with the `db_key` specified was found in the Recovery Appliance.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Action: Specify an existing `db_key`.

ORA-64822: Log file cannot be accepted from DBID %s on Tkey %s.
Cause: While performing redo shipping for the specified database, an error was detected that prevented the creation of the log file.

Action: Refer to the other errors in the error stack to determine the cause of the problem.

ORA-64823: Unable to move individual backup piece with BP_KEY %s; backup piece is not ready to move.
Cause: An attempt was made to move an individual backup piece, but the move backup operation encountered the resource busy issue for this backup piece. The backup piece still needed to replicate.

Action: Retry the move command.

ORA-64825: Handle %s already exists for database %s.
Cause: Another backup piece with the same handle already exists on the Recovery Appliance.

Action: For control file autobackup pieces in a Data Guard configuration, use the command `RMAN CONFIGURE AUTOBACKUP FORMAT` to ensure each database in a Data Guard configuration has a unique autobackup format string.

ORA-64826: filesize mismatch in %s for handle %s: catalog size=%s, but container size=%s
Cause: An internal error caused by an inconsistency in the size of the specified file maintained by two different components of the Recovery Appliance.

Action: Contact Oracle Support Services and provide trace and alert files..

ORA-64827: Database %s is not registered and therefore cannot be suspended.
Cause: A database that was not registered could not be suspended.

Action: Choose a database that has been registered.

ORA-64828: Database %s is not registered and therefore cannot be resumed.
Cause: A database that was not registered could not be resumed.

Action: Choose a database that has been registered.

ORA-64829: Database %s is not suspended and therefore cannot be resumed
Cause: A database that was not suspended could not be resumed.

Action: Use the `suspend.db` API to suspend the database.

ORA-64830: Incremental backup for database %s is not up-to-date.
Cause: Can be caused by a undersized appliance or indication of problem with backup for particular database.

Action: Run capacity planning report and review sizing of storage location as well as CPU, memory and disk utilization of components to seeif appliance is using more than 80% of resources for extended period of time. Also check for other incidents that may help describe the problem.If the cause is still not clear open a service request for Support.

ORA-64831: No outstanding backups to request for specified group.
Cause: Backup request group is either empty or has requested all backups.

Action: No action is necessary. This is an informational message only.

Copyright © 2014, 2026, Oracle and/or its affiliates.

ORA-64832: Error interacting with remote Recovery Appliance: %s; replication server: %s; database: %s.
Cause: An error was encountered during communication or conversationwith a remote Recovery Appliance for the specified database with the specified replication server. This is typically due toa network communication failure, missing or incorrect wallet,credentials, missing or incorrectly ordered add_db and grant_db_access calls. The error could also indicate an issueduring the conversation with the remote Recovery Appliance.

Action: Check the incident log for active failures with a component type of REMOTERA. Resolve stated issue and rerun the command.

ORA-64833: Insufficient free space at this time is stopping backup acceptance.
Cause: The Recovery Appliance is experiencing an extreme shortage of free space. To prevent failure of redo shipping, new backups are temporarily being rejected.

Action: Retry operation after purging has caught up and free space has grown to a level that can accommodate incoming backups.
If free space does not recover, consider removing databases or reducing recovery windows on the Recovery Appliance.

ORA-64834: Failed to replicate backup piece with bp_key %s and handle %s.
Cause: An error was encountered while backup pieces was being replicated onto configured replication server.

Action: The subsequent error will identify the error that needs to be addressed.

ORA-64835: Recovery Appliance API error %s.
Cause: An error was detected during the execution of an API requirest.

Action: Address the conditions that are identified by the attached message.

ORA-64837: In storage destination %s, tried to create %s GB, but could only create %s GB.
Cause: During a `create_storage_location` or `update_storage_location`, an attempt was made to create container files in the specified storage location, but only a portion of the space could be added to the storage location.

Action: Verify that ASM has available space in the storage destination. If the available space is less than expected, use `asmcmd` to determine whether unwanted files have been created in the storage destination. Unwanted files may appear when a previous operation started to create a storage location and its files were not properly removed.

ORA-64838: Task %s of type %s waiting on event %s for %s seconds.
Cause: The specified task remains stuck waiting for the specified event. This is only a warning message.

Action: If these errors persist, contact Oracle Support Services.

ORA-64839: Task %s of type %s interrupted after waiting on event %s for %s seconds.
Cause: The specified task remained stuck waiting for the specified event. Its process was stopped and restarted.

Action: If these errors persist, contact Oracle Support Services.

Copyright © 2014, 2026, Oracle and/or its affiliates.

ORA-64840: Illegal attempt to change a ZDLRA owned sbt_parms (%s); use procedure parameter (%s) instead.
Cause: An attempt was made to change an internal ZDLRA system parameter through `sbt_parms`.

Action: Remove the offending `sbt_parms` and instead use the specified procedure parameter to the `update_replaction_server` API.

ORA-64841: replication wallet_path cannot be NULL.
Cause: The replication server `wallet_path` cannot be NULL.

Action: Specify a valid `wallet_path`.

ORA-64842: Illegal attempt to manage a replication server using generic APIs. Use %s instead.
Cause: An attempt was made to cahnge a replication server using an API that does not support replication.

Action: Use replication specific APIs as given in the error message.

ORA-64843: Database with DBID %s is already registered in the catalog, tenant key %s. Deleting unregistered database %s.
Cause: An attempt was made to add a database with a DBID that is already registered in the catalog. Add two databases with the same DBID, such as primary and standby, to the same tenant is not permitted.

Action: Don't register databases with identical DBIDs to the same Recovery Appliance, same tenant. Because this database is already registered, you can connect to the catalog and perform your activities.

ORA-64844: Only %s GB out of %s GB reserved space is available to provision database %s.
Cause: An attempt was made to assign an initial `reserved_space` value to an `autotuned_reserved_space` database which was added with no initial `reserved_space` value, but there was insufficient unreserved space in the storage location to make the assignment.

Action: Remove unneeded databases from the Recovery Appliance to free up additional space that may be reserved. Manually assigning a lower `reserved_space` value to the database using the `update_db` API may also address the situation.

ORA-64845: Database %s must have a defined reserved_space value.
Cause: An attempt was made to assign a database without a `reserved_space` value to a protection policy with `autotune_reserved_space` set to `NO`.

Action: Specify a `reserved_space` value for the database using the `update_db` API or assign it a different protection policy with `autotune_reserved_space` set to `YES`.

ORA-64846: Database %s is growing too fast.
Cause: The `recovery_window_space` for an autotuned database has been growing at a rate that is unsupported. Instead of assigning the computed `reserved_space` setting to the database, a `reserved_space` is assigned that is constrained by the maximum permitted growth rate and the maximum reserved space permitted by the database's protection policy.

Action: If the `reserved_space` is constrained by the database's protection policy, assign the database to a protection policy that will support its required size. If the additional growth is valid, manually assign a `reserved_space` to the database using the `update_db` API.

Copyright © 2014, 2026, Oracle and/or its affiliates.

ORA-64847: Storage location %s has no available space for autotuning reserved_space.
Cause: While autotuning the `reserved_space` settings within a storage location, the storage location ran out of unassigned space.

Action: Increase the size of the storage location, remove databases from the storage location, or decrease `recovery_window_goal` settings for databases within the storage location.

ORA-64848: Database %s has an estimated size of %s GB, but has a disk reservation of %s GB.
Cause: The estimated size of a databse exceeds the amount of space reserved for it within the storage location. The database is in danger of having its recovery window truncated if there is space pressure within its storage location.

Action: Increase the `reserved_space` assigned to the database.

ORA-64849: Storage location %s requires %s GB, but has only %s GB.
Cause: The sum of the estimaged sizes of the database within a storage location is greater than the total space of the storage location.

Action: Increase the size of the storage location, remove databases from the storage location or decrease `recovery_window_goal` settings for databases within the storage location.

ORA-64850: Only one of the following parameters must be specified (%s)..
Cause: Multiple parameters provided where only one is legal.

Action: Only specify one parameter from the list per request.

ORA-64851: A full backup of database %s from the past %s days is missing needed for archival backup.
Cause: While creating archival backup, level 0 backups of one or more data files were missing in the past `max_redo_to_apply` days.

Action: Query the `RA_SBT_TEMPLATE_MDF` view to determine the data files for which backups are missing. Increase `max_redo_to_apply` if older backups are present or take new backups of those data files.

ORA-64852: Missing archive log backup of database %s needed for archival backup.
Cause: Missing archive log backups while creating archival backup.

Action: Perform backup of archive log(s) which were not backed up.

ORA-64853: COPYALL replication feature not supported by downstream Recovery Appliance.
Cause: The COPYALL replication feature requires both Recovery Appliances to be running a version that support the feature. The downstream Recovery Appliance version is too old.

Action: Upgrade your downstream Recovery Appliance to a newer release.

ORA-64854: Database %s exceeds the maximum reserved space limit of %s GB.
Cause: A database violated the `maximum_reserved_space` limit specified by its protection policy.

Action: Assign the database to a protection policy that has a larger `maximum_reserved_space` limit.

Copyright © 2014, 2026, Oracle and/or its affiliates.

ORA-64855: Unable to copy piece on %s because it is not recognized as a backup piece.
Cause: A database violated the `maximum_reserved_space` limit specified by its protection policy.

Action: Assign the database to a protection policy that has a larger `maximum_reserved_space` limit.

ORA-64857: OK4POOL: database %s, datafile %s, bpkey %s, rejected encrypted backup.
Cause: TDE Encrypted backups with RMAN encryption or compression will not be virtualized, unless `RA_FORMAT=TRUE`.

Action: Either (a) remove RMAN database encryption and compression, and perform an INCREMENTAL LEVEL 0 backup of the encrypted datafiles, or (b) if you want to use RMAN encryption or compression, set `RA_FORMAT=TRUE` when RMAN channels are allocated.

ORA-64858: Attribute set in use by existing SBT job template.
Cause: Not allowed to delete an attribute set this is currently used by an existing SBT job template.

Action: Delete the SBT job template and then try deleting the attribute set.

ORA-64859: %s
Cause: .

Action:

ORA-64860: Insufficient space for `compliance_hold` in database %s.
Cause: The first block of the specified file does not contain information produced by `DBMS_BACKUP_RESTORE`. Either this file is not a backup piece or the first block of the backup piece is corrupt.

Action: Specify a different file name and retry the operation.

ORA-64861: unable to restore for `compliance_hold` time.
Cause: The time stamp specified for a `compliance_hold` is not within a valid disk restore range.

Action: Choose a time for which the database is restorable.

ORA-64862: Database %s requires an additional %s GB space for `recovery_window_compliance`.
Cause: The `reserved_space` assigned to the database is not large enough to guarantee that the space specified by `recovery_window_compliance` will not be purged.

Action: Increase the `reserved_space` setting for the targeted database or specify a smaller `recovery_window_compliance`.

ORA-64863: cannot perform %s request with backups under %s restrictions.
Cause: The API that was attempted would violate `compliance_hold`, `keep_compliance`, `recovery_window_compliance` restrictions, or SBT immutability.

Action: Allow the restrictions to lapse before performing the API request.

ORA-64864: SBT libraries are needed to support `guaranteed_copy` in policy %s.
Cause: At least one SBT library with the guaranteed attribute are needed to support the `guaranteed_copy` attribute in a protection policy..

Copyright © 2014, 2026, Oracle and/or its affiliates.

Action: Create an SBT library that has the guaranteed attribute before retrying the current operation.

ORA-64865: Library %s is needed to maintain immutable backups for database %s.
Cause: The library is currently referencing immutable backups which precludes the library from being modified by the current operation.

Action: Wait until backups stored in the library are no longer referenced by the designated database.

ORA-64866: The setting for %s conflicts with the setting for %s.
Cause: One or more of the specified settings conflicts with another setting provided in this request or a prior request.

Action: Resolve the conflict and try again.

ORA-64867: Backup set %s is not permitted to have its KEEP UNTIL TIME reduced or eliminated.
Cause: An attempt was made to change the KEEP UNTIL TIME of a backup set, but the Recovery Appliance KEEP_COMPLIANCE protection policy prevents this action.

Action: None. The KEEP UNTIL TIME for this bacup set cannot be reduced.

ORA-64868: Only RMAN encrypted backups are supported on this Recovery Appliance.
Cause: The Recovery Appliance has the SECURE_MODE protection policy set to YES for this database. This policy requires all backup data be encrypted with RMAN.

Action: Use RMAN with Transparent Data Encryption while creating your next backup.

ORA-64869: Unencrypted redo is not allowed for database %s.
Cause: The Recovery Appliance has the SECURE_MODE protection policy set to YES for this database. This policy requires all redo sent from the database to be encrypted.

Action: First alter the LOG_ARCHIVE_DEST settings to turn on Transparent Data Encryption. Then use RMAN to encrypt and backup the existing archived logs.

ORA-64870: Online redo shipping for DBID %s cannot be performed because the Recovery Appliance is preparing for planned maintenance.
Cause: The Recovery Appliance was unable to automatically receive archive logs from the indicated database because the Recovery Appliance will be undergoing maintenance operations in the near future.

Action: Retry the operation after the Recovery Appliance has been restarted.

ORA-64871: Database %s is already a COPYALL_BACKUPS replication destination from Recovery Appliance %s.
Cause: For the specified database, the downstream Recovery Appliance was already a COPYALL_BACKUPS destination from a different Recovery Appliance. COPYALL_BACKUPS can be specified from only one upstream Recovery Appliance.

Action: Move the database to a replication policy that does not have COPYALL_BACKUPS defined.

ORA-64872: Exceeded throttle_db_channel_limit %s.
Cause: Multiple RMAN jobs are running simultaneously and have exceeded the allotted channel count.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Action: Check the setting of config parameter `throttle_db_channel_limit` then check the number of channels used by your RMAN jobs.
Also check for errant jobs performing backup. It is also possible a nightly data file backup job is running at the same time an hourly archived log backup job is running. The solution is to either increase `throttle_db_channel_limit` or stagger the RMAN jobs so they do not run at the same time. Or you might simply use fewer channels in one or both jobs.

ORA-64873: Encryption information with id %s not found for backup piece %s.
Cause: The required encryption information was not fetched by the Recovery Appliance which prevented the backup piece from being validated or re stored.

Action: Contact Oracle Support Services and provide the necessary trace files.

ORA-64874: Database %s is already registered.
Cause: The specified database is already registered on the downstream Recovery Appliance. `COPYALL_BACKUPS` replication requires the database to be unregistered .

Action: Either run `DELETE_DB` on the downstream followed by the `ADD_DB` and `GRANT_DB_ACCESS` or move the database to a replication policy that does not have `COPYALL_BACKUPS` defined.

ORA-64875: Unable to run API command at this due to %s.
Cause: The Recovery Appliance is in a state in which the current API command is not permitted to run.

Action: Retry the API command when the specified operation completes.

ORA-64876: Found unknown DBID for polling policy %s.
Cause: Found some polling files that are not associated with an active database in a protection policy that uses the polling policy.

Action: Ensure that all databases are in the active state and that polling tasks run after databases are added to the polling policy. Ensure that the unknown databases are in a protection policy that uses the polling policy. To find the list of files that have failed, look for files with status `DBID NOT IN POLICY` in the view `ra_polling_files`.

ORA-64877: Some polling files have timed out for polling policy %s.
Cause: One or more polling files are incomplete and have exceeded the timeout period for further processing.

Action: Recreate the polling files in the polling directory. To find the list of files that have timed out, look for files with status `INCOMPLETE FILE` in the view `ra_polling_files`.

ORA-64878: Illegal attempt to add/update %s to tenant %s.
Cause: (a) The user is in a tenant other than `DEFAULT_TENANT`.
(b) An incorrect add/update was attempted of the user or the database to the `DEFAULT_TENANT`.
(c) An add/update of the database has been made to `REP_TENANT`.

Action: Retry the `register_tenant_user`/`add_db` with the desired tenancy or `update_db`/`add_db` to a tenancy other than `REP_TENANT`.

ORA-64879: Backups cannot exist on both upstream and downstream Recovery Appliances for database %s.
Cause: The specified database has backups on both the upstream and the downstream Recovery Applicance. `COPYALL_BACKUPS` replication requires backups to exist on one Recovery Appliance only..

Copyright © 2014, 2026, Oracle and/or its affiliates.

Action: Either run on the downstream Recovery Appliance `DELETE_DB` followed by `ADD_DB` and `GRANT_DB_ACCESS`. Or, on the upstream Recovery Appliance move the database to a replication policy that does not have `COPYALL_BACKUPS` defined.

ORA-64880: Restore point already exists.
Cause: The restore point name already exists. A restore point name must be unique.
Action: Either use a different name or delete the existing restore point on the Recovery Appliance with the same name.

Action: Either use a different name or delete the existing restore point on the Recovery Appliance with the same name.

ORA-64881: The provided restore_until_scn/restore_until_time is not in the restore range for database %s.
Cause: The `restore_until_scn/restore_until_time` should be in the restore range for the database.

Action: Retry the API using a restore_until_scn/restore_until_time that lies in the restore range for the database.

ORA-64882: Update %s is not allowed because one or more databases are associated with the user %s."
Cause: One or more databases is still associated with the user.

Action: Revoke_db_access for the specified user and retry the command.

ORA-64883: Update %s is not allowed because the replication server is not in a paused state."
Cause: The database is associated with a replication server that is not in a paused state.

Action: Pause the replication server then retry updating the database on the upstream and downstream.

ORA-64884: TLS certificate needs update.
Cause: TLS certificate is expiring or is expired.

Action: Replace expiring or expired TLS certificate with a new valid one.

ORA-64885: Online redo shipping for DBID %s cannot be performed because recovery from an instance failure is in progress.
Cause: The Recovery Appliance was unable to automatically receive archive logs from the indicated database because the Recovery Appliance is still recovering logs that were active when an instance went down.

Action: Retry the operation after waiting for all `INSTANCE_CHANGE` tasks to complete.

ORA-64886: Failed to create a container file with free space of %s GB which is less than the minimum file size of %s GB.
Cause: The creation of a container file failed due to free space being less than the minimum required file size as specified in the error message. ".

Action: Ensure there is enough free space available to create the container file, or contact support.

Copyright © 2014, 2026, Oracle and/or its affiliates.

ORA-64887: shutdown_recovery appliance timed out; continuing as abort_recovery_appliance.
Cause: The timeout interval for `shutdown_recovery_applicance` expired prior to the recovery appliance being quiesced. No more time will be spent waiting for activity to complete. Instead, all remaining activity on the recovery appliance will be aborted.

Action: No action is necessary. This is an informational message only.

ORA-64889: Downstream Recovery Appliance does not have enough unreserved space for COPYALL replication of database %s.
Cause: The database on the Downstream Recovery Appliance uses `autotune_reserved_space` to allocate reserved space, but does not have enough unreserved space to receive the `COPYALL_BACKUPS`

Action: Free up reserved space from other databases, put the downstream data base into a protection policy without `autotune_reserved_space`, or add additional storage on the downstream Recovery Appliance.

ORA-64890: Recovery Appliance must be upgraded to the latest version
Cause: The Recovery Appliance received a request from a newer version of the Smart Data Protection Agent.

Action: Upgrade the Recovery Appliance to the latest version to ensure optimal support for all versions of the Smart Data Protection Agent.

ORA-64891: Smart Data Protection Agent reported error %s with message %s
Cause: The Smart Data Protection Agent reported an error to the Recovery Appliance during execution of the rcv fetch task command.

Action: Review the error code and message details to diagnose and resolve the issue.

ORA-64892: Missing required configuration parameters: %s
Cause: One or more mandatory Smart Data Protection Agent configurations are not configured in the Recovery Appliance, preventing backup script generation.

Action: Review the error message for listed configurations and provide values for the missing Smart Data Protection Agent settings in the Recovery Appliance.

ORA-64895: The additional reserved space of %s GB required for recovery window compliance exceeds the protection policy limit of %s GB for database %s under protection policy %s.
Cause: Reserved space for the required recovery window compliance could not be auto tuned, because it would exceed the protection policy maximum limit for databases under the protection policy.

Action: Retry after increasing `max_reserved_space` setting for the protection policy.

ORA-64896: Insufficient space in storage location %s to accommodate required increase of reserved space.
Cause: Storage location does not have sufficient allocatable space to accommodate the increased reserved space.

Action: Retry after performing any of the following:

- Increase storage location autotune space limit, if there is space available under the storage location.

Copyright © 2014, 2026, Oracle and/or its affiliates.

- Add more storage to the storage location and then increase storage location autotune space limit.
- Reduce `recovery_window_goal` for protection policies with autotune enabled.
- Reduce `max_reserved_space` for databases under protection policies which have autotune enabled.
- Remove databases from the protection policy.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Glossary

alert

In Cloud Control, an indicator that a particular metric condition has been encountered. For example, an alert might indicate that a metric threshold has been reached.

attribute set

A set of parameters set at the job level while copying Recovery Appliance backups to tape. Attribute sets are created as part of a media manager library for each drive associated with this library.

Auto Service Request (ASR)

A product feature that automatically opens service requests when specific Recovery Appliance hardware faults occur. ASR detects faults in the most common server components, such as disks, fans, and power supplies.

automated delta pool space management

The set of operations in which a Recovery Appliance determines which blocks are no longer needed, and then deletes them. Specifically, space management includes:

- Determining which backups (both in a Recovery Appliance storage location and on tape) are obsolete or expired based on the disk recovery window goal and SBT retention policy
- Deleting unneeded blocks from the Recovery Appliance storage to meet the disk recovery window goal and reserved space parameters configured for each protected database
- Optimizing the delta pools to improve performance of restore operations

backup copy policy

An attribute of a protection policy that determines whether the Recovery Appliance must ensure that new backups are replicated or copied to tape before deletion.

backup ingest

The automated stage in which a Recovery Appliance scans a backup that was sent by a protected database. The Recovery Appliance decomposes the backup into smaller sets of blocks, writes the blocks into the appropriate storage location, and indexes the backups.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Indexing includes inserting rows into the Recovery Appliance metadata database to describe the physical location of every block.

backup mode

The database mode (also called *hot backup mode*) initiated when you issue the `ALTER TABLESPACE ... BEGIN BACKUP` or `ALTER DATABASE BEGIN BACKUP` statement before taking an online backup. You take a tablespace out of backup mode when you issue the `ALTER TABLESPACE ... END BACKUP` or `ALTER DATABASE END BACKUP` statement.

backup polling directory

A file system directory on shared storage, located outside the Recovery Appliance, that is a destination for backup pieces and archived redo log files from a protected database. The Recovery Appliance polls the directory at specified intervals, retrieves any found backup data, and then processes and stores the data.

backup polling policy

An optional Recovery Appliance object that defines a storage area where a client database will place backups without interacting directly with the Recovery Appliance. The polling policy defines the file system path to the storage and how often it is searched for new backups.

backup reception

The stage in which a protected database sends a backup over the network to a Recovery Appliance, but before the Recovery Appliance has indexed the backup.

backup window

The amount of time that it takes for a backup to complete.

block change tracking

A database option that causes Oracle AI Database to track data file blocks affected by each database update. The tracking information is stored in a block change tracking file. When block change tracking is enabled, RMAN uses the record of changed blocks from the change tracking file to improve incremental backup performance by only reading blocks known to have changed, instead of reading whole data files.

cascaded replication

A configuration in which a downstream Recovery Appliance also serves as an upstream Recovery Appliance for a Recovery Appliance further downstream.

Cloud Control

Oracle Enterprise Manager Cloud Control is Oracle's enterprise cloud management solution. It enables you to monitor and manage the complete Oracle IT infrastructure from a single

Copyright © 2014, 2026, Oracle and/or its affiliates.

console. The core components of the architecture include the Oracle Management Agent, Oracle Management Service, Oracle Management Repository, Enterprise Manager for Zero Data Loss Recovery Appliance plug-in, and Enterprise Manager Cloud Control Console.

copy-on-write snapshot

After a third-party storage snapshot is taken, and when the first change occurs on a storage block, the array copies the before-image block to a new location on disk. The snapshot maintains the before-image block for the snapshot and the new block for the active version of the database.

delta pool

A set of data file blocks from which a virtual full backup is constructed. Each separate data file backed up to a Recovery Appliance has its own separate delta pool. The delta pools reside in the delta store.

delta pool optimization

The automatic tracking and reorganizing of the delta pools. As old blocks are deleted and new incremental backups arrive for updated data files, the blocks in a backup can become less contiguous. This state can degrade the performance of restore operations. Recovery Appliance automatically reorganizes the blocks to maintain contiguity during ordinary maintenance and validation.

delta push

The transfer of backups and changes from protected databases to the Recovery Appliance. This solution consists of two operations that run on each protected database: real-time redo transport, and the incremental-forever backup strategy.

delta store

The total Recovery Appliance storage that is used to store client backup data. The delta store contains all data file and archived redo log backups.

disk recovery window goal

The interval in which a point-in-time recovery must be possible using only disk backups. For example, if the recovery window goal is 15 days, and if it is noon on April 25, then the goal is the ability to perform point-in-time recovery to any time on or after noon on April 10. The goal, which is specified for each protection policy, is not a hard limit.

downstream Recovery Appliance

In a Recovery Appliance replication topology, the downstream Recovery Appliance receives replicated data from an upstream Recovery Appliance.

Copyright © 2014, 2026, Oracle and/or its affiliates.

enrolling a database

The process of enabling a specific Recovery Appliance to receive backups from a protected database. Enrolling involves adding the protected database (`DBMS_RA.ADD_DB`), granting access to this database to a Recovery Appliance User Account (`DBMS_RA.GRANT_DB_ACCESS`), and registering this database in the virtual private catalog (RMAN `REGISTER DATABASE` command).

fast recovery area

An optional disk location that you can use to store recovery-related files such as control file and online redo log copies, archived redo log files, flashback logs, and RMAN backups.

guaranteed copy

An optional setting of a protection policy that indicates that every backup must be copied to tape or replicated. Recovery Appliance cannot purge backups from the storage location until the operation succeeds. If tape or replication does not keep up, then the Recovery Appliance may reject new backups.

incremental-forever backup strategy

The strategy in which an initial level 0 backup is taken to the Recovery Appliance, with all subsequent incremental backups occurring at level 1. The Recovery Appliance creates a virtual full backup by combining the initial level 0 with subsequent level 1 backups.

media manager library

This is the media management library that manages tape backup job s. This library consists of attribute sets for each of its contained drives and defines storage parameters that apply to tape backup jobs.

media management software

The media management software is the middleware between the Recovery Appliance and the tape. It controls and manages the copying of backups from the Recovery Appliance to tape.

Recovery Appliance uses Oracle Secure Backup as its media management software and comes preconfigured with it.

one-way Recovery Appliance replication

The simplest form of the Recovery Appliance replication, in which one upstream Recovery Appliance sends backups to one downstream Recovery Appliance.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Oracle Configuration Manager

A tool that collects and uploads configuration information from Oracle homes in your environment. If you log a service request, then the configuration data enables Oracle Support Services to provide better service.

Oracle Enterprise Manager Cloud Control

See Cloud Control.

protected database

A client database that backs up data to a Recovery Appliance.

protection policy

A group of attributes that control how a Recovery Appliance stores and maintains backup data. Each protected database is assigned to exactly one protection policy, which controls all aspects of backup processing for that client.

real-time redo transport

The continuous transfer of redo changes from the SGA of a protected database to a Recovery Appliance. Real-time redo transport enables RMAN to provide a recovery point objective (RPO) near 0. Typically, RMAN can recover to within a second of the time when the failure occurred. Protected databases write redo entries directly from memory to the Recovery Appliance as they are generated.

reconciling

In Recovery Appliance replication, the process by which a Recovery Appliance receives metadata from the Recovery Appliances that are immediately downstream.

Recovery Appliance

Shortened name for Zero Data Loss Recovery Appliance. Recovery Appliance is an Oracle Engineered System specifically designed to protect Oracle databases. Integrated with RMAN, it enables a centralized, incremental-forever backup strategy for hundreds to thousands of databases across the enterprise, using cloud-scale, fully fault-tolerant hardware and storage.

Recovery Appliance User Account

A user account that is authorized to connect to, and request services from, Recovery Appliance. Every Recovery Appliance user account is an Oracle Database user account on the Recovery Appliance metadata database, and the owner of a virtual private catalog. When RMAN backs up a protected database, it connects to the recovery catalog with the Recovery Appliance user account credentials.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Recovery Appliance administrator

The administrator who manages a Recovery Appliance. Typical duties include creating and adding databases to protection policies, managing storage space, managing user accounts, configuring tape backups and the Recovery Appliance replication, and monitoring the Recovery Appliance.

Recovery Appliance Backup Module

An Oracle-supplied SBT library that RMAN uses to send backups of protected databases over the network to the Recovery Appliance. The library must be installed in each Oracle home used by a protected database.

The module functions as an SBT media management library that RMAN references when allocating or configuring a channel for backup to the Recovery Appliance. RMAN performs all backups to the Recovery Appliance, and all restores of complete backup sets, using this module.

Recovery Appliance metadata database

The Oracle database that runs inside of the Recovery Appliance. This database stores configuration data such as user definitions, protection policy definitions, and client database definitions. The metadata database also stores backup metadata, including the contents of the delta store.

Recovery Appliance replication

A configuration in which one Recovery Appliance receives backups, and then forwards them to another Recovery Appliance. The forwarder is the upstream Recovery Appliance, and the receiver is the downstream Recovery Appliance.

Recovery Appliance schema

The schema on the Recovery Appliance metadata database owned by the `RASYS` user. The schema is the super-set of the recovery catalog schema, and contains additional metadata used internally by Recovery Appliance to manage backups.

Recovery Appliance storage location

A set of Oracle ASM disk groups within Recovery Appliance that stores backups. A storage location can be shared among multiple protected databases. Every Recovery Appliance contains the default Recovery Appliance storage location named `DELTA`.

recovery point objective (RPO)

The data-loss tolerance of a business process or an organization. The RPO is often measured in terms of time, for example, five hours or two days worth of data loss.

Copyright © 2014, 2026, Oracle and/or its affiliates.

recovery window

A setting that defines how long the Recovery Appliance maintains tape backups in its catalog for recovery purposes.

recovery window goal

The time interval within which a protected database must be recoverable to satisfy business requirements. For each protected database in a protection policy, the Recovery Appliance attempts to ensure that the oldest backup on disk is able to support a point-in-time recovery to any time within the specified interval (for example, the past 7 days), counting backward from the current time.

redo staging area

For Recovery Appliance installations that enable real-time redo transport recovery, the Recovery Appliance storage destination for redo streams transmitted by protected databases. The Recovery Appliance converts the redo streams into archived redo log files, which it then converts to backup pieces and writes to a storage location.

replication user account

Oracle requires that you create a replication user account exclusively for use with Recovery Appliance replication, and that you create a unique replication user account for each upstream appliance within the organization.

Oracle recommends that the replication user account takes the form of `REPUSER_FROM_[ZDLRA_DB_NAME` or `ZDLRA_DB_LOCATION]`.

For example, if two Recovery Appliances have the `DB_UNIQUE_NAME` of ZDLRA1 and ZDLRA2, then the replication user accounts could be `REPUSER_FROM_ZDLRA1` and `REPUSER_FROM_ZDLRA2`. Or if those same Recovery Appliances were in Florence and Vienna, then the replication user accounts could be `REPUSER_FROM_FLORENCE` and `REPUSER_FROM_VIENNA`

The replication user account **should not** be used as a regular VPC user employed by protected databases to connect and send backups to the Recovery Appliance.

A database user account on the downstream Recovery Appliance that upstream Recovery Appliances will use to authenticate with this downstream Recovery Appliance.

reserved space

The minimum amount of disk space in the Recovery Appliance that is reserved for use by one protected database to meet its disk recovery window goal. The reserved space cannot be consumed by any other protected database. In general, the Recovery Appliance ignores reserved space settings until it is under space pressure, when it uses these settings and recovery window goals to determine which backups to purge.

retention policy

The length of time, expressed as a window of time extending backward from the present, that backups are kept on a SBT device. Backups may be kept longer than the specified window

Copyright © 2014, 2026, Oracle and/or its affiliates.

because they are kept long enough to guarantee that point-in-time recovery is possible to any point within the retention policy window.

RMAN recovery catalog

A set of metadata views residing in the Recovery Appliance metadata database.

SBT

System Backup to Tape. This term specifies a backup device type, typically either a tape device or Recovery Appliance. RMAN supports channels of type disk and SBT.

tape backup job

The operation that copies Recovery Appliance backups to tape based on the defined properties such as the associated media manager library, attribute set, backup type, and run-time window. This repeatable job can be scheduled to run immediately after being created or at a later specified time.

third-party storage snapshot

A set of pointers, managed by a third-party storage device, to storage blocks (*not* Oracle blocks) that existed when the snapshot was created. The device maintains a snapshot on the same storage array as the original data. The device only creates new versions of storage blocks when the snapshot perceives that they have changed.

unprotected window threshold

The user-specified maximum amount of data loss for protected databases that are subject to a protection policy. For example, a specified threshold of 5 minutes for protection policy `SILVER` means that every database protected by `SILVER` can lose no more than 5 minutes of data.

upstream Recovery Appliance

In a Recovery Appliance replication topology, the Recovery Appliance that is replicating backups to another Recovery Appliance.

virtual full backup

A complete database image as of one distinct point in time, maintained efficiently through the indexing of incremental backups from a protected database. The virtual full backups contain individual blocks from multiple incremental backups. For example, if you take a level 0 backup on Monday with SCN 10000, and if you take an incremental level 1 backup on Tuesday with SCN 11000, then the Recovery Appliance metadata database shows a virtual level 0 backup current to SCN 11000.

Essentially, virtual full backups are space-efficient, pointer-based representations of physical full backups as of the point-in-time of an incremental backup. When a restore operation is

Copyright © 2014, 2026, Oracle and/or its affiliates.

required, the delta store re-creates a physical full backup from the appropriate incremental backup SCN.

virtual private catalog

A subset of the metadata in a base RMAN recovery catalog to which a database user account is granted access. Each restricted user account has full read/write access to its own virtual private catalog.

Zero Data Loss Recovery Appliance

See Recovery Appliance.

Zero Data Loss Recovery Appliance Backup Module

See Recovery Appliance Backup Module.

Copyright © 2014, 2026, Oracle and/or its affiliates.

Index

A

accessing Recovery Appliance reports
  basic tasks, *6*
accessing the Recovery Appliance Home page, *2*
accessing the Recovery Appliance Storage Locations page, *4*
ALLOW_BACKUP_DELETION, *1*
Archival Backup to Tape and Cloud, *1*
archival backups, *25*, *26*, *3*
archive log
  grouping, *4*, *1*
archive to cloud, *1*, *2*
  add cloud storage, *13*
  Cloud_Key, *11*
  Cloud_User, *11*
  create TDE master keys, *11*, *19*
  credential wallet, *9*
  enable encryption keystore, *11*
  encryption keystore, *9*
  immutable bucket, *16*
  install OKV client software, *10*
  OCI objects, *11*
  Oracle Key Vault, *3*
  SBT_JOB_TEMPLATE, *17*
  ZFS OCI Object Storage, *17*
archive-to-cloud
  archive log grouping, *4*, *1*
asynchronous redo transport services, *8*
attribute sets, SBT
  creating, *16*, *17*
  default values, *8*
  deleting, *21*
  editing, *21*
Auto Service Request (ASR), *2*
automated delta pool space management, *7*, *16*

B

Backup and Redo Failover
  configuring, *2*
Backup Anywhere
  Mode for Data Guard, *24*, *25*
  Request Mode for Data Guard, *25*
BACKUP command, *3*
backup ingest phase, *6*
backup mode, *4*
backup polling
  directories, *19*, *16*
  how backups are processed, *20*
  locations, *17*, *20*
  policies, *10*, *13*, *19*, *3*, *7*, *16*
  stages, *20*
backup strategies
  full backups to third-party deduplicating appliance, *3*
  incremental backups and RECOVER COPY, *2*
  incremental-forever, *6*
  third-party snapshots, *4*
  weekly full and daily incremental, *1*
backup windows, reducing, *13*
backups
  archive to cloud, *1*
backups, RMAN
  archival, *26*
  archive to cloud, *3*
  copying to tape, *1*
  migrating, *1*
  replicating, *1*, *12*
  virtual full, *15*
BI Publisher reports, *6*
block change tracking, *1*

C

Capacity Planning Details report, *2*
Capacity Planning Summary report, *2*
cascaded replication, *7*
channel device
  failover to downstream, *17*
client software, *8*
cloud
  archive backups to, *1*, *3*, *10*, *19*
  Oracle Key Vault, *3*
Cloud Control administrator, *1*
Cloud Control for Recovery Appliance, *3*, *2*, *4*, *1*
  accessing the Oracle Secure Backup domain, *12*
  accessing the Replication page, *23*
  alerts, *3*
  BI Publisher reports, *6*

Copyright © 2014, 2026, Oracle and/or its affiliates.

Cloud Control for Recovery Appliance *(continued)*
centralized management of Recovery Appliance, *16*
Create Protection Policy page, *4*, *10*
displaying all Recovery Appliances, *1*
monitoring tools, *1*
Recovery Appliance Reports page, *4*
Storage Locations page, *4*
updating protected database properties, *7*
user accounts, *4*
cloud storage
archive to cloud, *10*
archive-to-cloud, *13*
immutable bucket, *16*
Cloud_Key
archive to cloud, *11*
Cloud_User
archive to cloud, *11*
compliance
DBMS_RA, *1*
racli add db_user, *1*
Compliance, *2*
Compliance Configuration, *1*
Compliance Hold, *6*
CONNECT CATALOG command, *5*, *2*, *3*
CONNECT TARGET command, *5*, *2*
convert tape to cloud
overview, *1*
copy-on-write snapshots, *4*
copy-to-tape jobs, *11*, *7*, *6*, *1*
managing, *28*
viewing status, *33*
copy-to-tape to cloud convertion, *1*
copying backups to tape, *1*
about pausing and resuming of, *6*
backup retention, *6*
basic tasks, *10*
components, *4*
overview, *1*
creating SBT media pools, *16*
credential wallet
archive to cloud, *9*

D

data files and delta pools, *15*
Data Guard, *24*, *25*
Backup Anywhere, *24*
replication, *25*
Data Guard Broker
failover to downstream, *21*
Data Security, *2*, *3*, *6*, *8*, *11*, *13*
database registration, *1*
DBMS_RA package, *3*
alternative to Cloud Control, *16*
DBMS_RA package subprograms, *1*
DBMS_RA.ADD_DB, *22*, *1*, *5*, *2*, *28*, *32*, *35*
DBMS_RA.ADD_REPLICATION_SERVER, *2*, *28*, *39*
DBMS_RA.COPY_BACKUP, *26*, *3*, *10*, *4*
DBMS_RA.CREATE_POLLING_POLICY, *5*, *5*, *17*
DBMS_RA.CREATE_PROTECTION_POLICY, *21*, *5*, *9*, *10*, *4*, *2*, *28*, *31*, *34*
DBMS_RA.CREATE_REPLICATION_SERVER, *2*, *28*, *38*
DBMS_RA.CREATE_SBT_ATTRIBUTE_SET, *10*, *17*, *4*
DBMS_RA.CREATE_SBT_JOB, *9*, *4*
DBMS_RA.CREATE_SBT_JOB_TEMPLATE, *25*
DBMS_RA.CREATE_SBT_LIBRARY, *9*, *16*, *4*
DBMS_RA.DELETE_PROTECTION_POLICY, *5*, *16*, *10*, *4*
DBMS_RA.DELETE_REPLICATION_SERVER, *28*
DBMS_RA.DELETE_SBT_ATTRIBUTE_SET, *10*, *4*
DBMS_RA.DELETE_SBT_JOB, *9*, *30*, *4*
DBMS_RA.DELETE_SBT_LIBRARY, *9*, *22*, *4*
DBMS_RA.GRANT_DB_ACCESS, *17*, *1*, *6*, *32*, *35*
DBMS_RA.MOVE_BACKUP, *26*, *10*, *4*
DBMS_RA.PAUSE_SBT_LIBRARY, *9*, *32*, *4*
DBMS_RA.QUEUE_SBT_BACKUP_TASK, *10*, *30*, *4*
DBMS_RA.REMOVE_REPLICATION_SERVER, *28*
DBMS_RA.RESUME_SBT_LIBRARY, *9*, *32*, *4*
DBMS_RA.REVOKE_DB_ACCESS, *3*
DBMS_RA.UPDATE_DB, *22*, *3*, *9*, *28*
DBMS_RA.UPDATE_PROTECTION_POLICY, *5*, *14*, *10*, *4*
DBMS_RA.UPDATE_SBT_ATTRIBUTE_SET, *10*, *22*, *4*
DBMS_RA.UPDATE_SBT_JOB, *9*, *29*, *4*
DBMS_RA.UPDATE_SBT_JOB_TEMPLATE, *29*
DBMS_RA.UPDATE_SBT_LIBRARY, *9*, *4*
DBMS_RS.ABORT, *5*
DBMS_RS.ABORT_RECOVERY_APPLIANCE, *5*
DBMS_RS.ADD_DB, *6*
DBMS_RS.ADD_REPLICATION_SERVER, *7*
DBMS_RS.CONFIG, *8*
DBMS_RS.COPY_BACKUP, *10*
DBMS_RS.COPY_BACKUP_PIECE, *11*
DBMS_RS.CREATE_ARCHIVAL_BACKUP, *13*
DBMS_RS.CREATE_POLLING_POLICY, *16*
DBMS_RS.CREATE_PROTECTION_POLICY, *16*
DBMS_RS.CREATE_REPLICATION_SERVER, *22*
DBMS_RS.CREATE_RESTORE_POINT, *21*
DBMS_RS.CREATE_SBT_ATTRIBUTE_SET, *23*
DBMS_RS.CREATE_SBT_JOB_TEMPLATE, *24*, *27*

Copyright © 2014, 2026, Oracle and/or its affiliates.

DBMS_RS.CREATE_SBT_LIBRARY, *27*
DBMS_RS.DELETE_DB, *28*
DBMS_RS.DELETE_POLLING_POLICY, *29*
DBMS_RS.DELETE_PROTECTION_POLICY, *30*
DBMS_RS.DELETE_REPLICATION_SERVER, *30*
DBMS_RS.DELETE_SBT_ATTRIBUTE_SET, *31*
DBMS_RS.DELETE_SBT_JOB_TEMPLATE, *31*
DBMS_RS.DELETE_SBT_LIBRARY, *31*
DBMS_RS.ESTIMATE_SPACE, *32*
DBMS_RS.GET_REDO_TRANSPORT_LAG, *33*
DBMS_RS.GRANT_DB_ACCESS, *33*, *39*, *44*
DBMS_RS.KEY_REKEY, *33*, *34*
DBMS_RS.MIGRATE_TAPE_BACKUP, *35*
DBMS_RS.MOVE_BACKUP, *35*
DBMS_RS.MOVE_BACKUP_PIECE, *37*
DBMS_RS.PAUSE_REPLICATION_SERVER, *39*
DBMS_RS.PAUSE_SBT_LIBRARY, *40*
DBMS_RS.POPULATE_BACKUP_PIECE, *40*
DBMS_RS.QUEUE_SBT_BACKUP_TASK, *41*
DBMS_RS.REMOVE_REPLICATION_SERVER, *42*
DBMS_RS.RENAME_DB, *42*
DBMS_RS.RESET_ERROR, *43*
DBMS_RS.RESUME_REPLICATION_SERVER, *44*
DBMS_RS.RESUME_SBT_LIBRARY, *45*
DBMS_RS.REVOKE_DB_ACCESS, *45*
DBMS_RS.SET_SYSTEM_DESCRIPTION, *46*
DBMS_RS.SHUTDOWN, *46*
DBMS_RS.SHUTDOWN_RECOVERY_APPLIANCE, *47*
DBMS_RS.STARTUP, *47*
DBMS_RS.STARTUP_RECOVERY_APPLIANCE, *48*
DBMS_RS.UPDATE_ARCHIVAL_BACKUP_KEEP, *49*
DBMS_RS.UPDATE_DB, *50*
DBMS_RS.UPDATE_POLLING_POLICY, *43*, *48*, *51*
DBMS_RS.UPDATE_PROTECTION_POLICY, *51*
DBMS_RS.UPDATE_REPLICATION_SERVER, *54*
DBMS_RS.UPDATE_SBT_ATTRIBUTE_SET, *55*
DBMS_RS.UPDATE_SBT_JOB_TEMPLATE, *56*
DBMS_RS.UPDATE_SBT_LIBRARY, *57*
DBMS_RS.UPDATE_STORAGE_LOCATION, *58*
DBMS_SCHEDULER.CREATE_JOB, *31*
delta pools, *15*, *15*
  automated space management, *7*
  for each data file, *15*
  optimization, *16*
delta push, *8*, *13*
  real-time redo transport, *14*
DELTA storage location, *19*, *10*
delta store, *13*, *15*
  delta pools, *15*
  virtual full backups, *15*
Disaster Recovery
  Data Guard Broker, *21*
  failover to downstream, *8*, *9*, *11*–*14*, *17*, *18*, *20*
  log_archive*, *23*
  Real-Time Redo Transport, *21*, *23*
  VPC user, *21*
disk recovery window goals, *10*, *12*, *16*, *18*, *2*, *4*, *1*, *7*

E

encrypted backups, RMAN, *15*
encryption
  HTTPS, *2*, *3*, *6*, *8*, *11*, *13*
encryption keystone
  archive to cloud, *11*
encryption keystore
  archive to cloud, *9*
Endpoint Group, *5*
Endpoints, *4*, *6*
enrolling protected databases, *1*
Enrollment Tokens, *7*
Enterprise Manager for Zero Data Loss Recovery Appliance plug-in
  See Recovery Appliance plug-in
expired backups, *7*

F

fast recovery areas, *18*
Fibre Channel, *10*
fleet
  racli add remote_syslog, *2*
fleet management, *2*

H

HADR, *23*, *25*
HTTPS, *2*, *3*, *6*, *8*, *11*, *13*
  racli add certificate, *8*
  racli alter network, *8*
  racli create certificate, *8*

I

Immutable Backups, *1*
immutable bucket
  cloud storage, *16*
incremental-forever strategy, *6*, *13*, *6*, *24*
  how it works, *14*

Copyright © 2014, 2026, Oracle and/or its affiliates.

K

KEEP_COMPLIANCE, *1*
Key Vault Server
  acquiring enrollment tokens, *7*
  associate wallet with endpoints, *6*
  dowload client software, *8*
  Endpoint Group, *5*
  Endpoints, *4*
  Wallet, *5*
keystone
  archive to cloud, *11*
keystore
  archive to cloud, *9*

L

legal hold, *6*
Legal Hold, *1*
libraries, SBT
  defined, *5*
LIST BACKUPSET command, *41*
log compression, *3*
log_archive*
  failover to downstream, *23*

M

master key, TDE, *11*, *19*
media manager libraries
  creating using Cloud Control, *13*
  deleting, *18*
media managers, *9*
media pools, SBT
  creating, *16*
  defined, *5*
migrating RMAN backups, *5*
monitoring Recovery Appliance, *6*
  Auto Service Request (ASR), *2*
  Incident Manager, *5*
  metric and collection settings, *4*
  performance, *7*
  with Cloud Control, *1*
  with Oracle Configuration Manager, *2*

O

OAP Publisher reports, *1*
obsolete backups, *7*
OCI objects
  archive to cloud, *11*
OKV client software, *8*
Oracle Analytics Publisher reports
  Capacity Planning Details, *2*
  Capacity Planning Summary, *2*
  Protected Database Chargeback Greatest, *2*
Oracle Analytics Publisher reports *(continued)*
  Protected Database Chargeback Least, *2*
  Protected Database Details, *2*
  Recovery Appliance Reports, *4*
  Recovery Window Summary, *2*
  System Activity, *3*
  Top 10 Protected Databases by Data Transfer, *3*
Oracle ASM, *12*
Oracle Configuration Manager, *2*
Oracle Enterprise Manager Cloud Control (Cloud Control)
  See Cloud Control
Oracle Key Vault
  archive to cloud, *3*
Oracle Scheduler,
  scheduling SBT jobs with, *30*
Oracle Secure Backup, *10*, *3*, *7*, *28*, *2*
  tape archival, *29*
  tape retrieval, *30*
Oracle Secure Backup domains
  accessing, *12*
Oracle wallets, *2*, *40*
  creating, *1*, *36*
  orapki utility, *37*
orapki utility, *37*

P

pausing and resuming tape copy operations
  about, *6*
protected database
  TDE master key, *19*
Protected Database
  failover to downstream, *14*
Protected Database Chargeback Greatest report, *2*
Protected Database Chargeback Least report, *2*
Protected Database Details report, *2*
protected databases, *6*, *2*
  access using DBMS_RA, *9*
  adding, *1*
  administrator, *1*
  configuring for replication, *40*
  enrolling with Recovery Appliance, *1*
  Recovery Appliance schema, *16*
  registering, *1*
  status reports, *2*
  updating properties using Cloud Control, *7*
protection policies, *6*
  about, *1*
  backup polling policy settings, *10*, *7*
  basic tasks, *6*
  benefits, *17*
  Cloud Control page, *4*
  copy-to-tape settings, *10*, *7*

Copyright © 2014, 2026, Oracle and/or its affiliates.

protection policies *(continued)*
creation, *5*
DBMS_RA procedures, *5*
definition, *9*
disk recovery window goals, *10*, *7*
for copying backups to tape, *5*
log compression, *3*
managing, *1*
overview, *9*, *2*, *3*
replication, *2*
replication server configurations, *11*, *8*, *39*
storage attributes, *21*
updating, *14*
Protection Policy
failover to downstream, *13*

R

RA_ACCESS view, *7*
RA_ACTIVE_SESSION view, *2*
RA_API_HISTORY view, *3*
RA_CONFIG view, *4*
RA_DATABASE view, *21*, *16*, *6*, *7*, *10*, *53*, *4*
RA_DATABASE_HISTORY view, *8*
RA_DATABASE_STORAGE_USAGE view, *10*
RA_DATABASE_SYNONYM view, *9*
RA_DB_ACCESS view, *10*, *10*
RA_DISK_RESTORE_RANGE view, *11*
RA_EM_SBT_JOB view, *10*
RA_EM_SBT_JOB_TEMPLATE view, *12*
RA_ENCRYPTION_INFO view, *13*
RA_HOST view, *53*
RA_INCIDENT_LOG view, *13*
RA_INCOMING_BACKUP_PIECES view, *14*
RA_PARTNER, *13*
RA_POLLING_FILES, *14*
RA_POLLING_POLICY view, *17*, *15*
RA_PROTECTION_POLICY view, *13*, *15*, *16*, *8*, *10*, *53*, *16*
RA_PURGING_QUEUE view, *23*, *17*
RA_RECOVERY_COMPLIANCE view, *18*
RA_REPLICATION_CONFIG view, *38*, *41*, *18*
RA_REPLICATION_DATABASE view, *19*
RA_REPLICATION_PAIR view, *20*
RA_REPLICATION_POLICY view, *21*
RA_REPLICATION_SERVER view, *53*
RA_REQUEST_BACKUP view, *23*
RA_RESTORE_RANGE view, *22*
RA_SBT_ATTRIBUTE_SET view, *10*, *23*
RA_SBT_JOB view, *10*, *24*
RA_SBT_LIBRARY view, *10*, *32*, *33*, *25*
RA_SBT_MIRROR view, *26*
RA_SBT_RESTORE_RANGE view, *26*
RA_SBT_TASK view, *34*, *27*
RA_SBT_TEMPLATE_MDF view, *28*
RA_SERVER view, *29*
RA_STORAGE_HISTOGRAM view, *29*
RA_STORAGE_LOCATION view, *30*
RA_STORAGE_LOCATION_HISTORY view, *30*
RA_TASK view, *31*
RA_TIME_USAGE view, *33*
RA_TIMER_TASK view, *33*
RASYS user account, *4*, *16*, *9*, *12*, *14*, *15*, *17*, *5*, *6*
RC_BACKUP_PIECE_DETAILS view, *53*
real-time redo transport, *8*, *2*, *3*, *1*
about, *13*
RECOVER COPY command, *2*
Recovery Appliance,
alerts, *4*
backup ingest phase, *6*
downstream Recovery Appliance, *8*
listing available Recovery Appliances, *1*
management through Cloud Control, *16*
migrating backups, *3*
monitoring, *6*, *1*
replication solution, *8*
roles, *1*
tape solution, *9*
validation, *12*
warnings, *4*
Recovery Appliance administration
separation of duties, *1*
tools, *2*
Recovery Appliance administrator, *1*
Recovery Appliance backup modules, *6*, *8*, *5*, *38*
Recovery Appliance environment, *1*
Recovery Appliance Home page
accessing, *2*
Recovery Appliance metadata database, *6*, *2*, *8*, *14*, *2*, *3*
Recovery Appliance migration
configuring with RACLI, *2*
Recovery Appliance partners, *1*
Recovery Appliance plug-in, *16*
Recovery Appliance replication, *8*, *7*, *8*, *30*, *6*, *4*, *1*, *12*, *1*
Airgap, *21*
backup anywhere, *32*
cascade mode, *34*
cascaded replication, *7*
configuring downstream Recovery Appliance, *30*
configuring using Cloud Control, *24*
configuring using DBMS_RA, *28*, *42*
configuring with RACLI, *15*
converting to RACLI management, *21*
examples, *5*
guidance, *14*
how it works, *30*, *2*
how RMAN restores backups, *4*
hub-and-spoke, *32*
one-way, *31*

Copyright © 2014, 2026, Oracle and/or its affiliates.

Recovery Appliance replication *(continued)*
overview, *1*
protection policies, *1*, *2*
read-only, *33*
reconciling, *31*, *3*
request_only mode, *33*, *10*
testing, *40*
upstream Recovery Appliance, *8*
Recovery Appliance replication>
bi-directional, *32*
Recovery Appliance Reports page, *4*
Recovery Appliance schema, *16*
Recovery Appliance service tiers, *17*, *2*, *5*, *6*, *9*, *12*, *8*
Recovery Appliance storage
backup polling locations, *17*
Cloud Control, *4*
DELTA storage location, *19*
guaranteed copy, *21*
locations, *3*, *10*, *15*, *17*, *5*, *7*
max_retention_window, *21*
recovery_window_goal, *21*
types, *17*
Recovery Appliance user accounts, *5*, *5*, *1*
Recovery Appliance workflow
planning, *2*
setup and configuration, *4*
recovery catalog, *6*, *11*, *3*, *14*, *17*, *31*
owned by RASYS, *5*
views, *5*, *10*
Recovery Manager (RMAN), *1*
recovery point objective (RPO), *5*, *8*
Recovery Window Compliance, *1*, *2*
Recovery Window Goal, *2*
Recovery Window Summary report, *2*
recovery window, SBT, *6*
RECOVERY_WINDOW_COMPLIANCE, *1*
Redo Transport
failover to downstream, *21*
REGISTER DATABASE command, *1*
Remote syslog, *2*
replication
Mode for HADR, *23*, *25*
Request Mode for Data Guard, *25*
replication server
failover to downstream, *12*
REPLICATION_DB, *13*
REPLICATION_SERVER, *13*
REPLICATION_USER, *13*
REPLICATION_WALLET, *13*
reports
Capacity Planning Details, *2*
Capacity Planning Summary, *2*
Protected Database Chargeback Greatest, *2*
Protected Database Chargeback Least, *2*
Protected Database Details, *2*
reports *(continued)*
Recovery Appliance Reports, *4*
Recovery Window Summary, *2*
System Activity, *3*
Top 10 Protected Databases by Data Transfer, *3*
request mode, *25*
reserved space, *3*, *4*
RMAN backups
archival, *25*
encrypted, *15*
handling obsolete and expired, *7*
lifecycle, *6*
migrating, *3*, *5*
RMAN commands
BACKUP, *3*
CONNECT CATALOG, *5*, *3*
CONNECT TARGET, *5*, *2*
LIST BACKUPSET, *41*
RECOVER COPY, *2*
REGISTER DATABASE, *1*
SWITCH, *2*
RMAN-encrypted backups, *15*

S

SBT jobs
creating using DBMS_RA, *25*
scheduling using Cloud Control, *30*
scheduling with Oracle Scheduler, *30*
viewing status using Cloud Control, *32*
SBT libraries, *6*, *8*, *30*
defined, *5*
managing using DBMS_RA, *21*
SBT media pools
creating, *16*
defined, *5*
SBT recovery windows, *6*
SBT retention periods, *12*
SBT_JOB_TEMPLATE
archive to cloud, *17*
Scheduler
See Oracle Scheduler
scheduling SBT jobs with Oracle Scheduler, *30*
Security, *1*
service tiers
See Recovery Appliance service tiers
service tiers, Recovery Appliance, *17*, *2*, *5*, *6*, *9*, *12*, *8*
shared walled
archive to cloud, *11*
SQL*Plus, *2*
SWITCH command, *2*
syslog, *2*
SYSMAN account, *4*
System Activity report, *3*

Copyright © 2014, 2026, Oracle and/or its affiliates.

T

tape
about pausing and resuming the copying of backups to, *6*
copying backups to, *1*
overview of operations on the Recovery Appliance, *3*
Recovery Appliance components for managing copying backups to, *4*
tape libraries, *9*
TDE master key
archive to cloud, *11*, *19*
third-party deduplicating appliances, *3*
third-party snapshots, *4*
TLS
Certificate, *2*
Client, *3*, *6*, *11*, *13*
Recovery Appliance, *8*
Top 10 Protected Databases by Data Transfer report, *3*
Transport
failover to downstream, *11*

U

user accounts
Cloud Control, *4*, *1*
RASYS, *4*
Recovery Appliance, *5*, *1*, *1*
user accounts *(continued)*
utilities
network_throughput_test.sh, *9*
rastat.pl, *7*

V

views, recovery catalog, *5*, *10*
virtual full backup, *6*, *15*
virtual full backups, *15*
copy-to-tape jobs, *11*
how they work, *15*
using replication, *9*
virtual private catalogs, *5*, *17*, *1*, *3*
VPC User
failover to downstream, *9*, *17*, *21*
redo transport, *21*

W

walled
shared, archive to cloud, *11*
Wallet, *5*, *6*

Z

Zero Data Loss Recovery Appliance
See Recovery Appliance
ZFS OCI Object Storage
archive to cloud, *17*

Copyright © 2014, 2026, Oracle and/or its affiliates.