

Oracle Linux

Installing and Configuring Oracle ASMLIB v3



G14969-10
June 2026



Oracle Linux Installing and Configuring Oracle ASMLIB v3,
G14969-10

Copyright © 2024, 2026, Oracle and/or its affiliates.

Documentation License

The content in this document is licensed under the [Creative Commons Attribution–Share Alike 4.0](https://creativecommons.org/licenses/by-sa/4.0/) (CC-BY-SA) license. In accordance with CC-BY-SA, if you distribute this content or an adaptation of it, you must provide attribution to Oracle and retain the original copyright notices.

Contents

Preface

1 About Oracle ASMLIB

ASMLIB Feature Support Matrix	1
Release Notes for Oracle ASMLIB 3.1.3	2
Release Notes for Oracle ASMLIB 3.1.1	2
Release Notes for Oracle ASMLIB 3.1	2
Release Notes for oracleasm-support 3.1	3
Known Issues	3
Release Notes for Oracle ASMLIB 3.0	4
Release Notes for oracleasm-support 3.0	4
Known Issues	5

2 Installing and Upgrading Oracle ASMLIB

Updating ULN Subscriptions or Downloading Packages	1
Installing Required Packages	2
Enabling or Restricting io_uring	2
Upgrading ASMLIB	3
Uninstalling ASMLIB	3

3 Configuring Oracle ASMLIB

Initializing ASMLIB Configuration	1
Configuring ASM I/O Filtering	2
Making Disks Available to ASMLIB	2
Checking ASMLIB Configuration Status	4
Validating ASM Disk Visibility Using a Discovery String	5
Querying ASM Disk Information	5

Preface

[Oracle Linux: Installing and Configuring Oracle ASMLIB v3](#) describes how to install and configure ASMLIB on Oracle Linux systems including Oracle Linux 8, Oracle Linux 9, and Oracle Linux 10.

1

About Oracle ASMLIB

Oracle ASMLIB is a recommended support library for Oracle Automatic Storage Management (ASM).

Using ASMLIB permits filtering out non-Oracle write I/Os and preventing them from overwriting ASM devices. This protection mechanism saves ASM disk groups from accidental overwrites of ASM data by non-Oracle entities. ASMLIB also supports thin provisioning and end-to-end data integrity protection, where the underlying storage provides this support.

ASMLIB Feature Support Matrix

ASMLIB provides a range of features that are dependent on facilities in the system kernel and on the version of the ASMLIB software installed. This table provides a reference that you can use to identify the appropriate minimum software version and platform and kernel combination that you need to take advantage of the different features in ASMLIB.

Table 1-1 Feature Support Matrix

Platform	Library to Kernel Interface	Read/Write I/O	End-to-End Data Integrity	I/O Filter	Automatic Block Size Selection	Thin Provisioning
Oracle Linux 10 with UEK 8	Built-in (io_uring)	3.0.0	3.1.0	3.0.0	3.1.0	3.1.0
Oracle Linux 10 with RHCK	Built-in (io_uring)	3.0.0	-	-	3.1.0	3.1.0
Oracle Linux 9 with UEK 8	Built-in (io_uring)	3.0.0	3.1.0	3.0.0	3.1.0	3.1.0
Oracle Linux 9 with UEK R7	Built-in (io_uring)	3.0.0	-	3.0.0	3.1.0	3.1.0
Oracle Linux 9 with RHCK	Built-in (io_uring)	3.0.0	-	3.0.0 ¹	3.1.0	3.1.0
Oracle Linux 8 with UEK R7	Built-in (io_uring)	3.0.0	-	3.0.0	3.1.0	3.1.0
Oracle Linux 8 with UEK R6	Built-in (oracleasm)	2.0.8	2.0.8	-	3.1.0 ²	-
Oracle Linux 8 with RHCK	RPM (kmod-redhat-oracleasm)	2.0.8	-	-	-	-
RHEL 10	Built-in (io_uring)	3.0.0 ³	-	-	3.1.0	3.1.0

Table 1-1 (Cont.) Feature Support Matrix

Platform	Library to Kernel Interface	Read/Write I/O	End-to-End Data Integrity	I/O Filter	Automatic Block Size Selection	Thin Provisioning
RHEL 9	Built-in (io_uring)	3.0.0 ⁴	-	3.0.0 ⁵	3.1.0	3.1.0
RHEL 8	RPM (kmod-redhat-oracleasm)	2.0.8	-	-	-	-
SLES 15	RPM (oracleasm-kmp-default)	2.0.8	2.0.8	-	3.1.1 ⁶	-
SLES 16	Built-in (io_uring)	3.1.1	3.1.1	3.1.1	3.1.1	3.1.1

¹ I/O Filter is available for x86_64 only.

² kernel-uek-5.4.17-2136.342.1 or later is required to support Automatic Block Size Selection on UEKR6.

³ io_uring interface marked Tech Preview.

⁴ io_uring interface marked Tech Preview.

⁵ I/O Filter is available for x86_64 only.

⁶ oracleasm-kmp-default-2.0.8-150600.16.13.2 or later is required to support Automatic Block Size Selection.

Release Notes for Oracle ASMLIB 3.1.3

Oracle ASMLIB is updated to version 3.1.3. The following notable changes are included:

- You can now specify the `io_uring` queue depth using the `ORACLEASM_MAX_QUEUE_DEPTH` option in `/etc/sysconfig/oracleasm`. This allows you to reduce the queue depth to accommodate slow storage devices.
- Previously, disks with a valid ASM disk label but a cleared label string would get discovered by ASMLIB. Now you can only discover disks which have a valid ASM disk label string.
- Fixed a rare bug that causes the wrong file descriptor to close if a disk device was removed by the kernel during the discovery process.

Release Notes for Oracle ASMLIB 3.1.1

Oracle ASMLIB is updated to version 3.1.1. The following notable changes are included:

- ASMLIB can now discover ASM disk labels on unpartitioned block devices.
- ASMLIB attempts additional retries when the kernel reports transient I/O errors.
- ASMLIB no longer lowers `maxio` for devices which report an invalid optimal I/O size.

Release Notes for Oracle ASMLIB 3.1

ASMLIB is updated to version 3.1. The following notable changes are included:

- End-to-End Data Integrity:** ASMLIB can use end-to-end data integrity protection when running on UEK R8 or later. This feature adds a layer of protection against data corruption

by attaching integrity metadata to each I/O. The attached integrity metadata can be validated by I/O controller and storage device on write operations, and by ASM on read operations.

End-to-end data integrity protection requires ASM disks to be placed on storage devices formatted with the T10 Protection Information, Type 1. Associated SCSI storage controllers must support Data Integrity Extensions (DIX). NVMe storage controllers must support the separate metadata buffer feature.

- **Automatic Block Size Selection:** ASM can now select whether to use the logical or the physical block size on a per-I/O basis. This feature makes it possible for ASM to migrate or access disks that were originally configured using different block sizes. In earlier releases, the block size selection was global and applied to all disks on a system.

To use per-I/O block size selection, the `ORACLEASM_USE_LOGICAL_BLOCK_SIZE` configuration option must be set to `false` and the Oracle Database must be patched to implement the per-I/O block size capability. See [Known Issues](#).

- **Thin Provisioning:** Unmap support is included, making it possible to reclaim unused disk space on thin-provisioned storage devices. This capability requires `KABI_V3` (UEK R7 or later). Use `oracleasm querydisk -i` or `oracleasm discover -l` to query whether an ASM disk supports thin provisioning.
- When using `KABI_V3` (UEK R7 or later) a universally unique identifier (UUID) is reported for those SCSI and NVMe storage devices that support it. Use `oracleasm querydisk -i` or `oracleasm discover -l` to get the UUID that uniquely identifies a storage device to ASM across all nodes in a cluster.

Release Notes for `oracleasm-support` 3.1

The following notable changes are included in `oracleasm-support` version 3.1:

- Disk labeling operations now inhibit writing ASM disks that are marked as being members of an ASM disk group. This is done to prevent inadvertently deleting or renaming ASM disks that are actively in use by other nodes in a cluster.
- Checks were added to the `oracleasm status` command to validate that the `oracleasm` service is fully operational. The checks include verifying that ASMLIB is installed and that the `io_uring` interface is accessible to the configured `ORACLEASM_UID/ORACLEASM_GID`.
- The `-i` option is added to the `oracleasm querydisk` command to print detailed information about the specified ASM disk.

The `-l` option is added to the `oracleasm discover` command to print detailed information about all ASM disks attached to the system.

The detailed information printed in each case includes device node, disk UUID, ASM disk group name, disk size, logical and physical block sizes, maximum I/O size, and whether the disk supports end-to-end data integrity and thin provisioning. Note that some information is only available when using `KABI_V3` interface (UEK R7 or later).

Known Issues

- The following Oracle Database patches must be installed to support mounting and migration of disks created using different ASM block sizes:
 - Patch 37347369 - VOTING DISK CREATED WITH AFD ON 512E DISK IN PAST DOES NOT WORK WITH ASMLIB

- Patch 37230154 - ASMCMD AFD_DECONFIGURE FAILS TO DELETE AFD RESOURCE
- Patch 39030808 - USING RDBMS.COMPATIBLE=10.1.0.0 WITH ASMLIB 3.1 ON 512E RESULTS IN ERROR.

If the Oracle ASM disk group attribute `compatible.rdbms` is earlier than 12.2.0.1.0, set `compatible.rdbms` to 12.2.0.1.0 or later. If changing `compatible.rdbms` isn't possible, apply patch 39030808 before upgrading to ASMLIB 3.1.

Release Notes for Oracle ASMLIB 3.0

Previous releases of Oracle ASMLIB supported several versions of interfaces implemented by the `oracleasm` driver over the years. The library picked the correct interface based on the `oracleasm` version reported by the kernel at runtime. Therefore, a single library binary could be used with various kernel releases.

Starting with ASMLIB version 3.0, another I/O submission interface is added to the library. The new I/O submission interface takes advantage of the high performance `io_uring` interface available in modern Linux kernels. If ASMLIB version 3.0 is loaded on a system that doesn't have an `oracleasm` driver loaded, is running a kernel that has `io_uring` enabled, and that supports a recent enough version of `io_uring`, the `io_uring` interface is used to submit I/O to the kernel instead of `oracleasm`.

The single ASMLIB version 3.0 binary handles all previous I/O submission interface versions in addition to `io_uring` and therefore no configuration changes are required when switching between kernels that don't have `io_uring` enabled and those that do.

The library automatically uses the appropriate interface for the running kernel.

Known Issues

- **Data integrity passthrough not supported with the `io_uring` interface.**

Data integrity passthrough isn't supported when using the `io_uring` interface, because of a kernel limitation. This issue might be resolved in a later kernel version.

Note that this limitation doesn't apply when running ASMLIB version 3.0 with the `oracleasm` driver on UEK R6.

- **ASMLIB version 3.0 for Arm is only supported with Oracle Database 19c**

Release Notes for `oracleasm-support` 3.0

The `oracleasm-support` package has been enhanced to work with version 3.0 of ASMLIB, which uses the generic `io_uring` interface to manage ASM disks on kernels that include this functionality. The updated `oracleasm-support` package continues to work with older kernels with backward compatibility. The command syntax remains the same irrespective of the kernel that's running.

I/O Filter: On systems running UEK R7 or Oracle Linux 9 with RHCK, `oracleasm-support` automatically adds an I/O filter to protect ASM disks against accidental overwrites. The filter rejects any write operations that aren't started by ASM and prevents writes to ASM disks by admin commands such as `dd` after disks have been instantiated. No new user level commands are required to manage the I/O filter map. If a disk device is found to have a valid ASM disk label, a filter map entry is automatically added.

The `oracleasm` configuration has a new parameter `ORACLEASM_CONFIG_MAX_DISKS`, which specifies the maximum number of ASM disks that can be used in the system. This parameter is used to calculate the size of I/O filter map.

I/O filtering depends on BPF (Berkeley Packet Filter) functionality within the kernel.

Known Issues

- **`oracleasm-support-3.0.0-7` or later required for use with Oracle ASM Dynamic Volume Manager (Oracle ADVM)**

You must install the `oracleasm-support-3.0.0-7` or later to use ASMLIB with Oracle ADVM on UEK R7 or later.

Patch 37405185 - ADD SUPPORT FOR ASMLIB V3 IN ADVM is also required in the Oracle Clusterware home directory.

2

Installing and Upgrading Oracle ASMLIB

ASMLIB is a support library for the Automatic Storage Management (ASM) feature of the Oracle Database. Oracle provides a Linux-specific implementation of this library. This document describes how to download, install, and upgrade the ASM library on Oracle Linux.

What do you need?

- A system with any of the following installed:
 - Oracle Linux 8
 - Oracle Linux 9
 - Oracle Linux 10
- Root administrator privileges on the host Linux system.

Note

If the host system is using Oracle Linux 7, see [Oracle Linux 7: Installing and Configuring Oracle ASMLIB](#).

Updating ULN Subscriptions or Downloading Packages

The `oracleasm-lib` and `oracleasm-support` packages are available on the Unbreakable Linux Network (ULN). You can also manually download the packages.

We recommend installing from ULN to keep the system updated.

- Subscribe to ULN Channels.

If you're using ULN, subscribe to the "Oracle ASMLIB" and "Oracle Linux Addons" ULN channels.

1. Sign in to <https://linux.oracle.com> with your ULN username and password.
2. On the Systems tab, from the list of registered systems, select the link name for the specified system.
3. On the System Details page, select Manage Subscriptions.
4. On the System Summary page, select the required "Oracle ASMLIB" and "Oracle Linux Addons" channels from the Available Channels list, then click the right-arrow (→) to move selected channels to the Subscribed Channels list.
5. Click Save Subscriptions.

- Download packages manually.

If you don't use ULN you can download the driver packages from the following resources, but you must keep them updated on the system when new patches are released:

- On Enterprise Linux 10: <https://www.oracle.com/linux/downloads/linux-asmlib-v10-downloads.html>

- On Enterprise Linux 9: <https://www.oracle.com/linux/downloads/linux-asmlib-v9-downloads.html>
- On Enterprise Linux 8: <https://www.oracle.com/linux/downloads/linux-asmlib-v8-downloads.html>

Installing Required Packages

The following steps describe how to install the ASM support tools, the ASM library, and the kernel driver (when applicable).

You must be subscribed to the ULN channels or have the packages downloaded before you proceed. See [Updating ULN Subscriptions or Downloading Packages](#) for more information.

1. Log into the system with an account that has administrator privileges.
2. Install the `oracleasm-support` and `oracleasm-lib` packages.

```
sudo dnf install oracleasm-support oracleasm-lib
```

3. Install the kernel driver if required.
 - a. On systems running UEK R7 or UEK 8, no driver is required to use ASMLIB because `io_uring` is already enabled in the kernel. You can restrict `io_uring` to processes run by a particular group id. See [Enabling or Restricting io_uring](#) for more information.
 - b. On systems running UEK R6 or earlier, the `oracleasm` kernel driver is already included and installed with the kernel.
 - c. On Oracle Linux 8 systems using RHCK, you must install the `oracleasm` driver:

```
sudo dnf install kmod-redhat-oracleasm
```

- d. On Oracle Linux 9 and Oracle Linux 10 systems using RHCK, you don't need to install the kernel driver, but you must enable `io_uring` in the kernel. See [Enabling or Restricting io_uring](#).

Enabling or Restricting `io_uring`

You might need to enable the `io_uring` interface on some systems, or you might decide to restrict it to processes run by a particular group.

The `io_uring` interface is used instead of the `oracleasm` kernel driver, when the system is running UEK R7 or later, or RHCK on Oracle Linux 9 or Oracle Linux 10. If the system is running RHCK on Oracle Linux 9 or Oracle Linux 10, you must either enable `io_uring` globally, or you can enable `io_uring` so that it's restricted to processes that are run by a particular group. `io_uring` is globally enabled in UEK R7 and UEK 8 by default, but you can edit the system configuration to restrict it to processes that are run by a particular group, if you prefer.

To edit the system configuration to enable or restrict `io_uring`, use the following procedure.

1. Edit or create `/etc/sysctl.d/io_uring.conf` to update the kernel configuration.
 - You can fully enable `io_uring` in the kernel by adding the following line to `/etc/sysctl.d/io_uring.conf`:

```
kernel.io_uring_disabled = 0
```

- Alternatively, you can restrict `io_uring` API access to Oracle Database processes by adding the following lines to `/etc/sysctl.d/io_uring.conf`:

```
kernel.io_uring_disabled = 1
kernel.io_uring_group = <GID>
```

Set `<GID>` to the numeric ID of the group specified when configuring ASMLIB with the `oracleasm configure` command.

2. Reload the system configuration.

If you have updated `/etc/sysctl.d/io_uring.conf`, reload the system configuration by running:

```
sudo sysctl -p /etc/sysctl.d/io_uring.conf
```

Upgrading ASMLIB

Update ASMLIB by using the `dnf update` command.

- Run the `dnf update` command to fully update the system.

```
sudo dnf update -y
```

We recommend that new ULN subscribers that have migrated from Red Hat update the `oracleasm-lib` package from ULN.

Uninstalling ASMLIB

Uninstall ASMLIB from a system.

1. Stop Oracle ASM and any running database instances on the node.

```
sudo srvctl stop instance -d db_unique_name-node node_name
sudo srvctl stop asm -node node_name
```

2. Stop the last Oracle Flex ASM instance on the node, by stopping the Oracle Clusterware stack.

```
sudo Grid_home/bin/crsctl stop crs
```

3. Stop the `oracleasm` service.

```
sudo systemctl disable --now oracleasm
```

4. Remove the `oracleasm-lib` and `oracleasm-support` packages.

```
sudo dnf erase oracleasm-lib oracleasm-support
```

3

Configuring Oracle ASMLIB

The following procedures are a guideline for the initial configuration of ASMLIB on Oracle Linux. For more information about Oracle Database, see <https://docs.oracle.com/en/database/oracle/oracle-database/>.

! Important

Changes to ASMLIB configuration don't take immediate effect. Stop the database and all related processes when reconfiguring ASMLIB. Changes usually take effect after the next system reboot.

Initializing ASMLIB Configuration

After installation, configure the ASMLIB software and scan for disks on boot by using the management utility, `/usr/sbin/oracleasm`.

You can read the ASMLIB configuration file at `/etc/sysconfig/oracleasm`, however we recommend that you always use `oracleasm configure` to change configuration parameters, so that this file is always correctly configured. The interactive (`-i`) option is typically used to configure the library for the first time.

- Run the configuration utility in interactive mode to initialize the configuration.

```
sudo oracleasm configure -i
```

If run for the first time, the `oracleasm configure` utility asks a series of questions, including which user id and group id to assign permission to use ASMLIB.

If the Oracle Database is configured to run as the `oracle` user and the `dba` group, the output looks similar to the following:

```
Configuring the Oracle ASM library driver.
```

```
This will configure the on-boot properties of the Oracle ASM library driver. The following questions will determine whether the driver is loaded on boot and what permissions it will have. The current values will be shown in brackets ('[]'). Hitting <ENTER> without typing an answer will keep that current value. Ctrl-C will abort.
```

```
Default user to own the driver interface []: oracle
Default group to own the driver interface []: dba
Start Oracle ASM library driver on boot (y/n) [n]: y
Scan for Oracle ASM disks on boot (y/n) [y]: y
Maximum number of disks that may be used in ASM system [2048]: 2048
Enable iofilter if kernel supports it (y/n) [y]: y
Writing Oracle ASM library driver configuration: done
```

- Enable and start the `oracleasm` service.

After you have finished configuring ASMLIB, enable and start the `oracleasm` service.

```
sudo systemctl enable --now oracleasm
```

Configuring ASM I/O Filtering

`oracleasm-support` includes an ASM I/O filtering feature that depends on BPF infrastructure support in the kernel. This feature is available in UEK R7 or Oracle Linux 9 with RHCK. When enabled, the I/O filter feature rejects any write operations that aren't started by ASM and prevents writes to ASM disks by admin commands such as `dd` after disks have been added to the ASM system.

See [Release Notes for oracleasm-support 3.0](#) for more information about the I/O filter feature.

1. Run the configuration utility to enable or disable I/O filtering.

By default, the I/O filter feature is enabled. Use the `oracleasm configure` command to disable or enable the I/O filter feature.

- a. Disable the I/O filter.

```
sudo oracleasm configure --iofilter n
```

- b. Enable the I/O filter.

```
sudo oracleasm configure --iofilter y
```

2. Run the configuration utility to set the maximum number of disk devices that ASMLIB can use with I/O filtering.

I/O filtering requires a mapping of the maximum number of disk devices that ASMLIB can use. The default value is 2048, but this value can be changed to any value, such as 4096, by running:

```
sudo oracleasm configure --maxdevs 4096
```

Making Disks Available to ASMLIB

Every disk that the Oracle Database accesses using ASMLIB must be labeled. This topic describes how to create an ASM disk label, verify it, and how to remove a label.

The following commands show how to scan disks, and create ASM disk labels. Instructions are also provided for viewing and querying disk labels and also for removing them.

1. Use the `oracleasm scandisks` command to scan all block devices attached to the system for ASM disk labels and make any ASM disks found available to ASM.

```
sudo oracleasm scandisks
```

Output might appear similar to the following:

```
Reloading disk partitions: done
Cleaning any stale ASM disks...
```

```
Setting up iofilter map for ASM disks: done  
Scanning system for ASM disks...
```

2. Use the `oracleasm createdisk` command to label a disk.

```
sudo oracleasm createdisk VOL1 /dev/sdg1
```

The following output might be displayed:

```
Writing disk header: done  
Instantiating disk: done
```

3. Use the `oracleasm listdisks` command to view existing disk labels.

```
sudo oracleasm listdisks
```

Output might appear similar to the following:

```
VOL1  
VOL2  
VOL3
```

4. Use the `oracleasm querydisk` command to check whether a disk device has a label.

```
sudo oracleasm querydisk /dev/sdg1
```

If the device isn't labeled as an ASM disk, the following output is displayed:

```
Device "/dev/sdg1" is not marked as an ASM disk
```

If the device is labeled as an ASM disk, output appears as follows:

```
Device "/dev/sdg1" is marked as an ASM disk with the label "VOL1"
```

You can also query an ASM disk label to see whether the label is valid, for example:

```
sudo oracleasm querydisk VOL1
```

5. Use the `oracleasm deletedisk` command to remove an ASM label from a disk.

```
sudo oracleasm deletedisk VOL1
```

The following output might be displayed:

```
Clearing disk header: done  
Dropping disk: done
```

When you have finished configuring disk availability, you can check that the disks are visible in ASM. See [Validating ASM Disk Visibility Using a Discovery String](#).

Checking ASMLIB Configuration Status

Use the `oracleasm status` command to show the status of ASMLIB configuration. This command can help identify issues and can show which features are enabled.

- Run `oracleasm status` to view the current configuration status.

```
sudo oracleasm status
```

The following example output is taken from a system running UEK R8:

```
Checking if the oracleasm kernel module is loaded: no (Not required with
kernel 6.12.0)
Checking if /dev/oracleasm is mounted: no (Not required with kernel 6.12.0)
Checking which I/O Interface is in use: io_uring (KABI_V3)
Checking if ASMLIB can be loaded: yes
Checking if io_uring is enabled: yes
Checking if io_uring is accessible to the configured DB user: yes
Checking if io_uring supports integrity passthrough: yes
Checking if ASM disks have the correct ownership and permissions: yes
Checking if ASM I/O filter is set up: yes
```

The following checks are performed:

- Check if the `oracleasm` kernel module is loaded: The kernel module is required for earlier kernels that don't include `io_uring`.
- Check if the `/dev/oracleasm` is mounted: When the `oracleasm` kernel module is used, a device node is configured and mounted. This action isn't required with kernels that include `io_uring`.
- Check which I/O interface is being used: in the case of a kernel that's using `KABI_V3` the `io_uring` interface is used, while a kernel using `KABI_V2` uses the `oracleasm` driver interface.
- Check that ASMLIB is installed and can be loaded.

Note that the following checks are only performed when `KABI_V3` is detected:

- Check if `io_uring` is enabled: On a kernel that includes `io_uring`, the `io_uring` feature must be enabled to use ASMLIB. See [Enabling or Restricting io_uring](#).
- Check that the `io_uring` is accessible to a process running with `ORACLEASM_UID/ORACLEASM_GID` credentials. See [Enabling or Restricting io_uring](#).
- Check whether end-to-end data integrity support is available through the `io_uring` interface (UEK R8).
- Check if ASM disks have correct ownership and permissions: Checks that any disk devices that are labeled for ASM use are owned by the user and group configured for ASM, and set when you initialized the configuration. See [Initializing ASMLIB Configuration](#).
- Check if ASM I/O filter is enabled and configured: On kernels that include the required BPF functionality, I/O filtering can be enabled and configured to protect ASM disks from accidental overwrites. See [Configuring ASM I/O Filtering](#).

Validating ASM Disk Visibility Using a Discovery String

ASM uses discovery strings to describe which of the labeled ASM disks attached to a system are available to the Oracle Database instance.

Use the `oracleasm discover` command to validate ASM discovery strings and view characteristics of the associated ASM disks.

- List all ASM disks on the system.

```
sudo oracleasm discover
```

The following output might be displayed:

```
Using ASMLib from /opt/oracle/extapi/64/asm/orcl/1/libasm.so
[ASM Library - Linux, version 3.0.0 (KABI_V3), Aug 26 2024 00:20]
Discovered disk: ORCL:DB1 [1044162 blocks (534610944 bytes), blksize
512/512, maxio 524288 bytes, integrity none]
Discovered disk: ORCL:DB2 [1044162 blocks (534610944 bytes), blksize
512/512, maxio 524288 bytes, integrity none]
Discovered disk: ORCL:VOL1 [1044162 blocks (534610944 bytes), blksize
512/512, maxio 524288 bytes, integrity none]
Discovered disk: ORCL:VOL2 [1044162 blocks (534610944 bytes), blksize
512/512, maxio 524288 bytes, integrity none]
```

- List all ASM disks whose labels begin with "VOL".

```
sudo oracleasm discover ORCL:VOL\*
```

The following output might be displayed:

```
Using ASMLib from /opt/oracle/extapi/64/asm/orcl/1/libasm.so
[ASM Library - Linux, version 3.0.0 (KABI_V3), Aug 26 2024 00:20]
Discovered disk: ORCL:VOL1 [1044162 blocks (534610944 bytes), blksize
512/512, maxio 524288 bytes, integrity none]
Discovered disk: ORCL:VOL2 [1044162 blocks (534610944 bytes), blksize
512/512, maxio 524288 bytes, integrity none]
```

Querying ASM Disk Information

You can query ASM disk information by either using the `oracleasm querydisk -i` or `oracleasm discover -l` commands.

- Use the `oracleasm querydisk -i` command to query detailed information about a specific ASM disk.

```
sudo oracleasm querydisk -i VOL1
```

The following output might be displayed:

```
ORCL:VOL1
Device: /dev/sda1 (8:1)
Device UUID: naa.333333330000007d0-part1
Disk Group: DBTEST
Capabilities: IO | UDID | NOGLOBAL | LOGICAL | IO_LOGICAL
Size: 130520 blocks (534609920 bytes)
Logical Block Size: 512 bytes
Physical Block Size: 4096 bytes
Max I/O Size: 128 blocks (524288 bytes)
Data Integrity: T10-DIF-TYPE1-CRC
Thin Provisioning: supported
Max TP I/O Size: 8388607 blocks (4294966784 bytes)
```

The output fields are as follows:

Device

The ASM disk's block device node.

Device UUID

Universally unique identifier of the underlying storage hardware (if supported).

Disk Group

Name of the ASM Disk Group that the ASM disk belongs to.

Capabilities

Features implemented by this ASM disk.

Size

Storage capacity of the ASM disk.

Logical Block Size

The logical block size of the underlying storage hardware.

Physical Block Size

The physical block size of the underlying storage hardware.

Max I/O Size

Maximum I/O size supported by the underlying storage hardware.

Data Integrity

Indicates whether the ASM disk supports end-to-end data integrity protection. A value of `T10-DIF-TYPE1-CRC` indicates that the device is formatted with T10 Protection Information, Type 1, and that exchanging protection information with the underlying storage hardware is enabled.

Thin Provisioning

Indicates whether the underlying storage hardware supports reclaiming unused space through an `Unmap` operation.

Max TP I/O Size

Indicates the maximum size of an `Unmap` operation.

- To query detailed information about all ASM disks attached to the system use the `-l` option when you run the `oracleasm discover` command.

```
sudo oracleasm discover -l
```

The information printed for each ASM disk matches the output of the `oracleasm querydisk -i` command.