Oracle® Exalogic Elastic Cloud

ExaLogs User's Guide

Release 1.3

E50008-03

October 2015

This document describes how to use ExaLogs, a tool for gathering logs, diagnostics, and configuration information from an Oracle Exalogic Elastic Cloud machine.

This document contains the following sections:

1 Getting Started with ExaLogs

This section contains the following topics:

If you are familiar with ExaLogs and want to quickly get started using it, go directly to Section 1.4, "Overview of the ExaLogs Task Flow."

1.1 What is ExaLogs?

ExaLogs is a command-line tool for gathering logs, diagnostics, environment/configuration information, and other data from the following components in an Exalogic physical or virtual configuration.

- Compute nodes (Dom0 in the case of a virtual configuration)
- Exalogic Control vServers hosting the following services (relevant for virtual configurations only):
  - Oracle Database
  - Oracle VM Manager
  - Enterprise Manager Ops Center Enterprise Controller
  - Enterprise Manager Ops Center Proxy Controllers
  - Oracle Virtual Assembly Builder (OVAB) Deployer

- ZFS storage appliance
- InfiniBand switches
- Guest vServers (relevant for virtual configurations only)

You can use ExaLogs to gather evidence either from the entire rack or from individual components.

1.2 Supported Platforms

For a list of the EECS releases that are supported for ExaLogs, see the My Oracle Support document ID 1912063.1.

> **Note:** Before running ExaLogs for an Exalogic machine that was upgraded to EECS 2.0.6.0.0, you must first synchronize the ECU configuration files with the current configuration of the machine, by running an ECU converter. The ECU converter is a tool that is included with ExaPatch. For more information about the ECU converter, see the *ExaPatch User's Guide*.

1.3 Installing ExaLogs

ExaLogs gets installed automatically when you install the ExaLogic Lifecycle (ELLC) toolkit. For the ELLC installation instructions, see My Oracle Support document ID 1912063.1.

The ExaLogs tool is available in the `/exalogic-lctools/bin` directory on the compute node on which you installed the ELLC toolkit.

1.4 Overview of the ExaLogs Task Flow

This section provides a high-level flow of the tasks you must perform to use ExaLogs. For information about how ExaLogs gathers diagnostics data from the components on an Exalogic machine, see Section 1.5, "Understanding How ExaLogs Works."

1. Specify the access credentials that ExaLogs must use for connecting to the components on the Exalogic machine, as described in Section 2.2, "Specifying Credentials for ExaLogs to Access Exalogic Components."
2. While running ExaLogs, if you want to provide information to ExaLogs about pre-discovered components (*targets*) on the Exalogic machine, perform the procedure described in Section 2.1, "Discovering Components on an Exalogic Rack."

   Alternatively, you can specify the IP addresses of the targets individually, directly in the ExaLogs command.
3. Run ExaLogs, as described in Section 3, "Using ExaLogs."

1.5 Understanding How ExaLogs Works

You can run ExaLogs from the following locations:

- For an Exalogic rack in a physical environment: The compute node on which the ZFS share containing ExaLogs was mounted when the Exalogic Lifecycle Toolkit was installed.

- For an Exalogic rack in a virtual environment: The compute node (dom0) on which the ZFS share containing ExaLogs was mounted when the Exalogic Lifecycle Toolkit was installed.

  You can also mount the ZFS share (in which ExaLogs is installed) on the vServer that hosts the Enterprise Controller and run ExaLogs from there.

When you run ExaLogs from the locations listed previously, it does the following:

1. Checks whether the local host—that is, the host on which ExaLogs is invoked—has sufficient space to store the diagnostics bundles.
2. Creates a directory structure on the local host, to store the diagnostics files.
3. For each target Exalogic component:
   a. Finds out the IP addresses for the component by using the user-specified IP address or the *discovered* IP address (for more information about discovery, see Section 2.1, "Discovering Components on an Exalogic Rack").
   b. Checks whether the component is running.
   c. Logs in to the component.
   d. Creates the staging directory `/tmp/exalogs-`*`date_stamp-hosttype_hostaddress`* on the component for which ExaLogs is gathering data.

      - `date_stamp` is in the `yymmdd_hhmm.ss_UTC` format.

      - `hosttype` is the type of host for which ExaLogs is gathering data.

      - `hostaddress` is the host name or IP address of the component.

      Example: `/tmp/exalogs-131210_1248.46_UTC-192.168.10.1`
   e. Creates a diagnostics directory under `/tmp/exalogs-`*`date_stamp-platform_host`*.
   f. Performs a series of operations to gather data about the component, and stores the data in `/tmp/exalogs-`*`date_stamp-hosttype_hostaddress`*.
   g. Generates a tarball of `/tmp/exalogs-`*`date_stamp-hosttype_hostaddress`* on the component.
   h. Downloads the tarball to the local host, extracts the tarball, and deletes the temporary tarball.
4. Generates the overall diagnostics bundle.
5. Generates a mini bundle for each component type.

Running ExaLogs in the *Remote* Mode

In the remote mode, ExaLogs copies the `exalogs.sh` script to each target on the ExaLogic rack, and the `exalogs.sh` script then runs within each target component to gather data about that component.

> **Note:** The remote mode is for use by Oracle field and support personnel only.

The remote mode is useful in the following scenarios:

- To run ExaLogs centrally against one or more racks.

- When the compute node on which ExaLogs is running does not have access to all the components of the rack.
- To ensure that no ExaLogs raw data remains after ExaLogs runs.

In the remote execution mode, ExaLogs does the following:

1. Checks whether the local and remote hosts have sufficient space to store the diagnostics bundles.
2. Creates a directory structure on the local host—that is, the compute node on which ExaLogs is invoked.
3. Checks whether the remote host is running.
4. Creates a directory on the remote host and copies `exalogs.sh` to that directory.

   If the remote host prompts for a password, ExaLogs does the following:

   - If the password is provided (by using the `--rexecpw` *`password`* option), ExaLogs attempts to log in by using the specified password.
   - Otherwise, ExaLogs attempts to log in by using the default (factory) passwords.
   - If all of the above fail, ExaLogs prompts the user for the password.
5. Runs the remote copy of `exalogs.sh` with all the relevant command-line arguments.
6. Performs the tasks described earlier for the local mode.
7. Downloads the diagnostics bundle from the remote host.
8. Removes the diagnostics bundle and the `exalogs.sh` script from the remote host.

2 Preparing to Use ExaLogs

This section contains the following topics:

2.1 Discovering Components on an Exalogic Rack

When you run ExaLogs, you can specify the IP addresses of the targets for which you want to gather evidence. Alternatively, you can *discover* all the components on the Exalogic rack and then pass the discovered data (in an XML file) to ExaLogs every time you run it. Note that discovery is a one-time operation. It is not necessary to run discovery again, except after hardware upgrades, after EECS release upgrades, and for sanity checking.

> **Note:** Do not use discovery XML files generated from previous releases of ExaLogs. Generate the discovery file afresh for your Exalogic machine by using the current release of ExaLogs.

This section describes how to discover components on an Exalogic machine. It contains the following topics:

- Section 2.1.2, "Specifying Access Credentials for ExaLogs Discovery"
- Section 2.1.3, "ExaLogs Discovery Usage Examples"

2.1.1 General Syntax for ExaLogs Discovery

The ExaLogs discovery tool is available in the `/exalogic-lctools/bin` directory on the compute node on which you installed the ELLC toolkit.

The following is the general syntax for the tool:

```
exalogs_discovery --access FILE|CRED[,...] --discover-from SOURCE[,...]
--discover-save FILE --discover-query
```

- The `--discover-from` option specifies the *source* from which the components on the Exalogic machine must be discovered.
  - If you have ECU configuration files that correctly reflect the current configuration of the Exalogic machine, Oracle recommends that you use those configuration files as the discovery source, by specifying `--discover-from` **`ecu`**. The tool looks for the ECU configuration files on the compute node specified by the `--access` option. On Exalogic physical environments, the tool looks in the `/var/exalogic/one-command/config` directory. On Exalogic virtual environments, the tool looks in the `/var/tmp/exalogic/ecu` directory.
  - If you do not have ECU configuration files that correctly reflect the current configuration of the Exalogic machine, you can specify `--discover-from` **`fabric`**. However, note that, with the **`fabric`** source, the following components are not discovered: Oracle VM Manager, Enterprise Controller, Proxy Controller, and Database components of Exalogic Control; the gateway switches; the Cisco switch; the PDUs; and the storage-node ILOMs.

    Note: Regardless of the discovery source you use, inspect the discovered data and ensure that it is correct and complete.

- The `--access` option specifies the credentials that must be used for accessing the source specified by the `--discover-from` option.
  - You can store the access credentials in a plain-text file (for example, `access.txt`), one credential string per line, and then specify the name of the file as the value of the `--access` option, as in the following example:

    ```
    --access access.txt
    ```

  - Alternatively, you can specify the access credentials in a comma-separated list directly on the command line, as in the following example:

    ```
    --access
    root:CN1:elcn01.example.com:::,root:CN01:192.168.20.1:::,root:EC01:10.162.5
    0.123:::
    ```

  For more information, see Section 2.1.2, "Specifying Access Credentials for ExaLogs Discovery."
- The `--discover-save` option specifies the full path and name of the XML file in which the output of the ExaLogs discovery command must be saved.

2.1.2 Specifying Access Credentials for ExaLogs Discovery

For the `--access` option, you can specify either a comma-separated list of credential strings or the full path and name of a plain-text file that contains the credential strings.

Each credential string must be in the following format:

user:*host_identifier*:*host*:*ip*:*password_file*:*private_key_file*

Table 1 describes the fields in the credential string. Note that invalid values in credential strings are ignored.

Table 1 Fields in the Access Credential String for ExaLogs Discovery

Field	Description
`user`	Specify the user that must be used to access the component/s specified in the credential string. **Note**: For a STIG-hardened Exalogic rack, the `user` field must contain a nonprivileged user (not `root`), as shown in the following example: `# sh ./exalogs_discovery --access` **`exalogic`**`:CN1:COMPUTENODE::: --discover-from ecu --discover-query --discover-save myrack.xml` In this example, in the access credential `exalogic:CN1:COMPUTENODE:::`, the user `exalogic` is a nonprivileged user that has been granted `sudo` privileges in the `/etc/sudoers` file on the target hosts on the rack.
`host_identifier`	Specify the component type, optionally suffixed with a numerical index that indicates the specific host. You can specify the following component types in the `host_identifier` field. Note that some names have synonyms. ■ `COMPUTENODE` (**synonyms:** `COMPUTE_NODE, COMPUTESERVER, COMPUTE_SERVER, CN, OVS, XEN, DOM0`) ■ `CTRLDB` (**synonyms:** `CTRL_DB, DB`) ■ `ENTERPRISECONTROLLER` (**synonyms:** `OPSCENTEREC, OPS_CENTER_EC, EC`) ■ `EXALOGICCONTROL` (**synonyms:** `ELCTRL, CTRLVM`) ■ `EXALOGICGENERICLINUX` (**synonyms:** `EXALOGICLINUX, LINUX`) **Note:** This type is generic and can be used for both physical Linux and virtual environments. ■ `EXALOGICGENERICSOLARIS` (**synonyms:** `EXALOGICSOLARIS, SOLARIS`) ■ `IBSWITCH` (**synonym:** `IB_SWITCH`) ■ `IBSWITCHGW` (**synonym:** `IB_SWITCH_GW`) ■ `IBSWITCHSPINE` (**synonym:** `IB_SWITCH_SPINE`) ■ `OVAB` ■ `OVMM` (**synonyms:** `OVMMANAGER, OVM_MANAGER`) ■ `PROXYCONTROLLER` (**synonyms:** `OPSCENTERPC, OPS_CENTER_PC, PC`) ■ `ZFSSTORAGEHEAD` (**synonyms:** `ZFS_STORAGE_HEAD, ZFS, ZFSSA, ZFS_SA`) **Example**: For the first compute node, the host identifier in the credential string would be `CN01`.
`host`	Specify either the host name or the IP address of the component.
`ip`	This field is deprecated. Leave it blank.
`password_file`	Specify the full path and name of the file that contains the plaintext password, and optionally the user, for the component. If no user is specified, ExaLogs assumes the password is for the `root` user. By default, the interactive mode is enabled. When no password file is found during discovery, a password prompt is displayed.

Table 1 (Cont.) Fields in the Access Credential String for ExaLogs Discovery

Field	Description
private_key_file	If passwordless SSH to the target component is enabled, specify the full path and name of the file that contains the private SSH key. **Note**: This field is relevant only for the ExaLogs discovery command. When you use the `--access` option with the ExaLogs tool, leave this field blank.

The following are examples of credential strings, constructed by using the fields described in Table 1:

Credential-String Example	Description
`root:CN1:elcn01.example.com:::`	Access the first compute node (with the host name, `elcn01.example.com`) as the `root` user, by using the default password.
`root:CN01:192.168.20.1:::`	Access the first compute node (with the IP address, `192.168.20.1`) as the `root` user, by using the default password.
`root:EC01:10.162.50.123:::`	Access the Enterprise Controller vServer (with the IP address, `10.162.50.123`) as the `root` user, by using the default password.
`:COMPUTENODE:::/tmp/pwd.txt:`	Access all compute nodes as the default user (`root`), by using the password specified in `/tmp/pwd.txt`.
`exalogic:COMPUTENODE:::`	Access all compute nodes as the `exalogic` user.
`exalogic:COMPUTENODE:::/tmp/pwd:`	Access all compute nodes as the `exalogic` user, by using the password specified in `/tmp/pwd.txt`.
`exalogic::192.168.20.1::/tmp/pwd:`	Access the target with the IP address `192.168.20.1` as the `exalogic` user, by using the password specified in `/tmp/pwd.txt`. ExaLogs automatically determines the type of the component.

2.1.3 ExaLogs Discovery Usage Examples

This section provides a few examples for generating a rack discovery file for ExaLogs.

The ExaLogs discovery tool is available in the `/exalogic-lctools/bin` directory on the compute node on which you installed the ELLC toolkit.

In all of the following examples, the discovered data is displayed on the console and saved in the file, `myrack.xml`.

- **Discovering components without specifying the host containing the discovery source**

  ```
  # sh ./exalogs_discovery --access root:CN1:192.168.20.15::: --discover-from ecu
  --discover-query --discover-save myrack.xml
  ```

  The tool looks for the ECU configuration files on the compute node specified by the `--access` option. On Exalogic physical environments, the tool looks in the `/var/exalogic/one-command/config` directory. On Exalogic virtual environments, the tool looks in the `/var/tmp/exalogic/ecu` directory. If you run this command from a compute node other than the one that contains the ECU files, a password prompt will be displayed.

- **Discovering components by providing a specific host as the discovery source**

Assume that you have defined the following host identifiers in the `access.txt` file: `CN1`, `CN2`, `CN3`. You can now specify one of these hosts in the `--discover-from` option, as in the following example:

```
# sh ./exalogs_discovery --access access.txt --discover-save myrack.xml
--discover-query --discover-from ecu:CN3
```

If you run this command from a compute node other than the one that contains the ECU files, a password prompt will be displayed.

- **Discovering components by providing a specific ECU directory on a specific host as the discovery source**

  By default on Exalogic physical environments, the tool looks in the `/var/exalogic/one-command/config` directory and on Exalogic virtual environments, the tool looks in the `/var/tmp/exalogic/ecu` directory. You can provide a specific directory on a specific host. Assume that you have defined hosts `CN01` and `CN02` in the `access.txt` file.

  You can now specify a directory on one of these hosts in the `--discovery-from` option, as in the following example:

  ```
  # sh ./exalogs_discovery --access access.txt --discover-save myrack.xml
  --discover-query --discover-from ecu:CN01:/tmp/ECU
  ```

  The tool looks for the ECU configuration files in the `/tmp/ECU` directory on the `CN01` host.

  Note: After generating the discovery XML file, you *must* inspect the data resulting from discovery. You can convert the XML data to a human- and script-friendly shell-variable format, by using the `--discovery-query-format shellvars` option, as shown in the following example:

  ```
  # sh ./exalogs_discovery --discover-from myrack.xml
  --discover-query --discover-query-format shellvars
  ```

  The following is an example of the output of this command:

  ```
  rack_0_id=AK00055531
  rack_0_name=el01
  rack_0_size=Eighth
  rack_0_ib_switch_00=10.10.54.193
  rack_0_ib_switch_00_Eth-admin=10.10.54.193
  rack_0_ib_switch_01=10.10.54.192
  rack_0_ib_switch_01_Eth-admin=10.10.54.192
  ...
  ```

  Consider saving the data formatted in the shell-variable format, for later use, by redirecting the output to a text file, as follows:

  ```
  # sh ./exalogs_discovery --discover-from myrack.xml
  --discover-query --discover-query-format shellvars > file_name
  ```

2.2 Specifying Credentials for ExaLogs to Access Exalogic Components

To access the components on an Exalogic rack, ExaLogs uses either SSH keys (if previously set up) or default passwords.

You can override the default passwords by using the environment variables listed in the following table:

Note: Alternatively, like for ExaLogs discovery, you can provide a passwords file to ExaLogs, by using the `--access` option, as described in Section 2.1.2, "Specifying Access Credentials for ExaLogs Discovery."

Environment Variable	Specifies the Password for the Component...	Specifies the Password for the User...
`ELCTRL_PASSWORD*`	vServer hosting the Oracle Virtual Assembly Deployer, EM Ops Center Enterprise Controller, Oracle VM Manager, and Oracle Database components of the Exalogic Control stack (relevant for only v2.0.6.x.x)	`root`
`OVAB_PASSWORD*`	vServer hosting the Oracle Virtual Assembly Deployer (relevant for only v2.0.4.x.x and earlier releases)	`root`
`EMOCEC_PASSWORD*`	vServer hosting the EM Ops Center Enterprise Controller (relevant for only v2.0.4.x.x and earlier releases)	`root`
`EMOCPC_PASSWORD*`	vServers hosting the EM Ops Center Proxy Controllers	`root`
`OVMM_PASSWORD*`	vServer hosting the Oracle VM Manager (relevant for only v2.0.4.x.x and earlier releases)	`root`
`OVMM_SERVICE_PASSWORD*`	Oracle VM Manager	`admin`
`CTRLDB_PASSWORD*`	vServer hosting the Oracle Database instance (relevant for only v2.0.4.x.x and earlier releases)	`root`
`ZFSSA_PASSWORD`	ZFS storage appliance heads	`root`
`IBSWITCH_PASSWORD`	NM2-GW switches	`root`
`OVS_PASSWORD*`	Compute nodes running Oracle VM Server	`root`
`LINUX_PASSWORD`	Compute nodes running Oracle Enterprise Linux	`root`
`SOLARIS_PASSWORD`	Compute nodes running Oracle Solaris	`root`

Note: The environment variables marked with an asterisk (*) are relevant only for Exalogic racks in a virtual configuration.

For information about passing environment variables to ExaLogs, see the UNIX documentation for the shell that you use. For example, for the Bash shell, you can use the following methods:

```
# export ELCTRL_PASSWORD=password
# sh ./exalogs options
```

Or

```
# env ELCTRL_PASSWORD=password ./exalogs options
```

Caution: Providing plain-text passwords in a script or at the command line is not a secure practice. Instead, set up key-based (password-less) access from the host on which you are running ExaLogs to all the targets from which you want to gather evidence, and secure the private key.

For information about setting up key-based access, see the standard UNIX documentation. For information about setting up SSH key-based access for the storage appliance, see the documentation at: http://docs.oracle.com/cd/E27998_01/html/E48433/configuration__services__ssh.html.

---

3 Using ExaLogs

This section contains the following topics:

- Section 3.1, "ExaLogs Command Syntax"
- Section 3.2, "ExaLogs Options"
- Section 3.3, "ExaLogs Targets"
- Section 3.4, "Directory Structure of the ExaLogs Diagnostics Bundle"
- Section 3.5, "ExaLogs Usage Examples"

3.1 ExaLogs Command Syntax

The ExaLogs tool is available in the `/exalogic-lctools/bin` directory on the compute node on which you installed the ELLC toolkit.

The following is the general syntax for the ExaLogs CLI commands:

```
exalogs option(s) [target(s)]
```

Note: For guest vServers or Linux compute nodes that have been hardened for compliance with STIG requirements, you must run ExaLogs as a nonprivileged user that has been granted `sudo` privileges with the `NOPASSWD` directive in the `/etc/sudoers` file on such STIG-hardened hosts. In addition, you must run the `exalogs` command after navigating to the `/exalogic-lctools/bin` directory or by specifying the full path to the command—that is, `/exalogic-lctools/bin/exalogs`.

- For information about the options that you can specify, see Section 3.2, "ExaLogs Options."

- For information about the supported targets, see Section 3.3, "ExaLogs Targets."

3.2 ExaLogs Options

The following table lists and describes the CLI options that you can use with ExaLogs.

Option	Purpose
`--help` `-h`	Display usage help for the ExaLogs command. **Example**: `exalogs` **`-h`**
`--access` *`FILE`*`|`*`CRED`*`[,...]` `-a` *`FILE`*`|`*`CRED`*`[,...]`	Access targets on the Exalogic rack by using the specified access credentials. **Example**: `exalogs` **`-a access.txt`** `--discover myrack.xml --all` For more information, see Section 2.1.2, "Specifying Access Credentials for ExaLogs Discovery."
`--clean` `-c`	Remove local diagnostics directories. Keep only the bundles. **Example**: `exalogs --discover myrack.xml --all` **`-c`**
`--debug` `-d`	Enable debug output. **Example**: `exalogs --discover myrack.xml --all` **`-d`**
`--debug2` `-d2 | -dd`	Enable detailed debug output. **Example**: `exalogs --discover myrack.xml --all` **`-dd`**

Option	Purpose

`--diagspath` *path*
`-p` *path*
	By default, when ExaLogs runs, it stores the diagnostics bundles in the current working directory. You can change the directory by using the `--diagspath` option.
	Example:
	`exalogs --discover myrack.xml --all` **`-p /test/exalogs/`**
	Note the following requirements and restrictions with regard to the directory that you can specify with the `--diagspath` option:
	▪ Ensure that the file system that contains the directory has at least 10 GB of space to store the diagnostics bundles.
	▪ *Do not* specify any of the following directories:
	`/export /home /media /misc /mnt /mnt1 /net /root /tmp /var /Users /volumes`
	▪ *Do not* specify a directory that is a subdirectory of any of the following directories:
	`/bin /boot /dev /etc /js /lib /lib64 /nlgc-setup /opt /proc /sbin /scmnt /selinux /srv /sys /tftpboot /usr /Applications /Developer /Library /Network /System`
	If you specify a subdirectory of one of these directories, ExaLogs *will* fail and display an error message.
	Note: ExaLogs does not manage the diagnostics bundles. After ExaLogs runs a few times, the file system might fill up with diagnostics bundles. You must manually remove any bundles that are no longer necessary. For the remote mode (`--rexec` option), you can use the `--remotecleanolder` option to remove data from older, interrupted runs of ExaLogs.

`--discover` *file*	Path to a discovery file, in XML or shell-variables format, that ExaLogs must use for identifying hosts. For information about generating the discovery file, see Section 2.1, "Discovering Components on an Exalogic Rack." **Example**: `exalogs` **`--discover myrack.xml`** `--all` Besides specifying `--discover` *xml_file*, if you provide a specific target (for example, `--ibswitch` or `--zfssa`), ExaLogs gives precedence to the specified target over the discovery file. In addition to `--discover` *xml_file*, for any target, if you specify an IP address on a network other than `eth-admin`, `EoIB-external-mgmt`, and `IPoIB-admin`, ExaLogs does not recognize that the user-specified and discovered IP addresses belong to the same target. In such cases, ExaLogs gathers evidence separately for the user-specified and discovered IP addresses, resulting in duplicate collection.
`--emoc-db-snapshot`	The EM Ops Center data model snapshot You can use this option to take a snapshot of the EM Ops Center data model. You must use this option with the `--elctrl`, `--emocec`, or `--all` targets. This snapshot is approximately 400 to 500 MB in size. This option should be used on only Exalogic racks running EECS 2.0.6 or later releases. Note that the ExaLogs-generated snapshots are purely for use by Oracle personnel during support engagements. The snapshots *must not* be used for restoring the EM Ops Center data model.
`--insane` `-i`	Gather more diagnostics. Note that this option can result in diagnostics bundles containing several GB of data **Example**: `exalogs --discover myrack.xml --all` **`-i`**

Option	Purpose
`--logs [true|false]`	Specify whether ExaLogs should gather data from log files (default: `true`). **Example**: `exalogs --discover myrack.xml --all` **`--logs false`**
`--noremoteclean`	Do not remove data on the remote host after transfer to the local host (useful for debugging). **Example**: `exalogs --discover myrack.xml --all` **`--noremoteclean`**
`--parallel`	Collect evidence from the target components in parallel. This is the default behavior. It results in better performance when compared with the `--serial` option. The verbose log is saved in only the `user.log` file. **Example**: `exalogs --discover myrack.xml --all` **`--parallel`**
`--prefer-network` *`network`* `-n` *`network`*	If you specify the `--discover` *`xml_file`* option, by default, ExaLogs uses the following networks, in the same order, for identifying hosts from the discovery file. ■ `eth-admin` ■ `EoIB-external-mgmt` ■ `IPoIB-admin` To force ExaLogs to use a particular network for identifying hosts from the provided discovery file, specify the preferred network by using the `--prefer-network` option. If you specify the host name or IP address when using a target, ExaLogs ignores the `--prefer-network` option and uses the network that corresponds to the host name or IP address. You can specify one of the following networks: ■ `eth-admin` ■ `eoib-external-mgmt` ■ `ipoib-admin` ■ `ipoib-default` ■ `ipoib-storage` ■ `ipoib-virt-admin` ■ `ipoib-ovm-mgmt` ■ `ipoib-vserver-shared-storage` **Example**: `exalogs --discover myrack.xml --all` **`-n ipoib-admin`**
`--remotecleanolder`	Remove all temporary ExaLogs files, including from previous ExaLogs runs, from every host for which evidence was gathered. When you run ExaLogs with the `--rexec` option, if you specify `--remotecleanolder` before `--rexec`, besides removing the temporary ExaLogs files on the hosts for which evidence was gathered, ExaLogs removes the temporary ExaLogs files on the host from which ExaLogs was invoked as well. **Example**: `exalogs --discover myrack.xml --all` **`--remotecleanolder`**

Option	Purpose
`--rexec` *`user@ip`*`[:`*`path`*`]` `-e` *`user@ip`*`[:`*`path`*	Run ExaLogs as the specified `user` on the specified remote IP address (or host name) at `path`. Note that `path` is optional. The default path is `/tmp/exalogs`. **Note**: The option is for use by only Oracle field and support personnel. For more information about remote execution, see "Running ExaLogs in the Remote Mode."
`--rexecpw` *`password_file`*	Specifies the password to be used when ExaLogs is executed remotely by using `--rexec`. **Note**: The remote mode is for use by only Oracle field and support personnel. For more information about remote execution, see "Running ExaLogs in the Remote Mode."
`--serial`	Collect evidence from the target components serially. This results in verbose output on the console, but the process is slower than when you specify `--parallel`. **Example**: `exalogs --discover myrack.xml --all` **`--serial`**
`--version` `-v`	Display the version of ExaLogs. **Example**: `exalogs -v`

For more examples, see Section 3.5, "ExaLogs Usage Examples."

> **Note:** Review the known issues listed in My Oracle Support document ID 1912063.1. For each issue, the document provides a description of the symptoms and causes, and solutions where available.

3.3 ExaLogs Targets

To run ExaLogs, you must specify the targets for which you want to gather evidence. You must specify at least one of the following targets.

> **Note:** When gathering evidence for guest vServers or Linux compute nodes that have been hardened for compliance with STIG requirements, the user being specified with the `--access` option must be a nonprivileged user that has been granted `sudo` privileges with the `NOPASSWD` directive to execute the `bash` shell in the `/etc/sudoers` file on such STIG-hardened hosts.

Target	Description
`--all`	All known components You *must* specify `--discover` *`xml_file`* as well.
`--ctrldb [`*`ip_or_host`*`]`	Oracle Database

Target	Description
`--elctrl [ip_or_host[,...]]`	All the Exalogic Control services found at the specified host names or IP addresses **Note**: ExaLogs detects duplicate `--elctrl` host names—both discovered and user-specified. For example, if you specify the IP address of the OVAB and OVMM services (`--ovab 192.168.20.15 --ovmm 192.168.20.15`) and the `--elctrl` target, then only a single ELCTRL target is considered.
`--emocec [ip_or_host]`	EM Ops Center Enterprise Controller
`--emocpc [ip_or_host[,ip_or_host]]`	One or both EM Ops Center Proxy Controllers
`--ibswitch [ip_or_host[,...]]`	One or more IB switches
`--linux [ip_or_host[,...]]`	One or more compute nodes running Oracle Enterprise Linux **Or** One or more guest vServers
`--ovmm [ip_or_host]`	Oracle VM Manager
`--ovab [ip_or_host]`	Oracle Virtual Assembly Builder
`--ovs [ip_or_host[,...]]`	One or more compute nodes running Oracle VM Server (Dom0s)
`--solaris [ip_or_host[,...]]`	One or more compute nodes running Oracle Solaris
`--targets ip_or_host[,...]`	One or more IP addresses or host names of *any* components on the Exalogic machine ExaLogs will attempt to detect the target type—IB switch, storage head, Linux compute node, and so on—for each of the specified IP addresses and host names. If ExaLogs cannot determine the target type for any of the specified IP addresses or host names, ExaLogs will not gather any data for such targets.
`--zfssa [ip_or_host[,ip_or_host]]`	One or both storage appliance heads

For the targets other than `--all`, you can specify the `--discover xml_file` option, *or* the host names (or IP addresses) of the individual targets, *or* both.

- If you specify only the `--discover xml_file` option, ExaLogs scans the discovery file for hosts that match the specified target. Additionally, in the case of the `--elctrl` target, ExaLogs probes the discovered hosts for any control-stack services running on the hosts.
- If you specify the `--discover xml_file` option *and* the IP addresses (or host names) of the targets, ExaLogs uses the preferred networks: `eth-admin`, `EoIB-external-mgmt`, and `IPoIB-admin`, in that order.

> **Note:** Review the known issues listed in My Oracle Support document ID 1912063.1. For each issue, the document provides a description of the symptoms and causes, and solutions where available).

3.4 Directory Structure of the ExaLogs Diagnostics Bundle

Directory Structure of the ExaLogs Diagnostics Bundle for an Exalogic Machine in a Virtual Configuration

```
exalogs-yymmdd_hhmm.ss_UTC
    STORAGE
         head1_ip_adress
             zfssa
                 config
                 logs
         head2_ip_address/...
             ...
    COMPUTENODE
        node1_ip_address
            ovs (subdirectories: config, state, test, and logs)
            sosreport.tar.bz2 (see Note at the end of this section)
            os (subdirectories: config, state, test, and logs)
            network (subdirectories: config, state, test, and logs)
        node2_ip_address/...
            ...
        ...
    CTRLSTACK
        ELCTRL_ip_address (the VM hosting the EC, OVMM, DB, and OVAB components)
            db (subdirectories: config, state, test, and logs)
            opscenter (subdirectories: config, state, test, and logs)
            os (subdirectories: config, state, test, and logs)
            ovab (subdirectories: config, state, test, and logs)
            ovmm (subdirectories: config, state, test, and logs)
            network (config, state, test, and logs)
        ELCTRL_PC1_ip_address
            opscenter (subdirectories: config, state, test, and logs)
            os (subdirectories: config, state, test, and logs)
            network (subdirectories: config, state, test, and logs)
        ELCTRL_PC2_ip_address/
            ...
    IBSWITCHES
         switch1_ip_adress
             nm2 (subdirectories: config, state, test, and logs)
             os (subdirectories: config, state, test, and logs)
             network (config, state, test, and logs)
         switch2_ip_address/...
            ...
```

Directory Structure of the ExaLogs Diagnostics Bundle for an Exalogic Machine in a Physical Configuration

```
exalogs-yymmdd_hhmm.ss_UTC
    STORAGE
         head1_ip_adress
             zfssa
                 config
                 logs
         head2_ip_address/...
            ...
    LINUX (SOLARIS in the case of an Exalogic machine running Oracle Solaris)
        node1_ip_address
            os (subdirectories: config, state, test, and logs)
            network (subdirectories: config, state, test, and logs)
```

```
        node2_ip_address/...
            ...
        ...
    IBSWITCHES
         switch1_ip_adress
             nm2 (subdirectories: config, state, test, and logs)
             os (subdirectories: config, state, test, and logs)
             network (config, state, test, and logs)
         switch2_ip_address/...
            ...
```

Directory Structure of the ExaLogs Diagnostics Bundle for Guest vServers

```
exalogs-yymmdd_hhmm.ss_UTC
    LINUX
        vserver1_ip_address
            os (subdirectories: config, state, test, and logs)
            network (subdirectories: config, state, test, and logs)
        vserver2_ip_address/...
            ...
        ...
```

Note:

- `sosreport.tar.bz2` is the output of the `sosreport` command, which ExaLogs invokes internally. For control-stack VMs, `sosreport` is disabled, by default, to avoid potential memory issues. You can force ExaLogs to run `sosreport` for the control-stack VMs, by using the `--insane` flag. When you do this, `sosreport` runs on the control stack VMs with the `rpm.rpmva` plugin disabled, because the plugin takes considerable time to run.
- The `oswatcher` service is configured and active on all Oracle VM Server nodes on an Exalogic machine running EECS 2.0.6 (or later releases) and on all Oracle Linux nodes. ExaLogs downloads the most recent detected `oswatcher` archive, and includes the `oswatcher.tgz` file in the diagnostics bundle at `COMPUTENODE/ip_address`.

3.5 ExaLogs Usage Examples

This section provides a few examples of ExaLogs usage.

- In the following example, ExaLogs is run from the vServer that hosts the Enterprise Controller component of the Exalogic Control stack, to collect diagnostics data for all the compute nodes, in parallel mode.

  ```
  [root@rack-elcontrol bin]# sh ./exalogs --discover myrack.xml --ovs
  ```

  For detailed information, similar to that displayed for the serial mode, see the log file (`exalogs.sh.user.log`). Note that the breakdown of the time it took for ExaLogs to run is not logged in the parallel mode.

- In the following example, ExaLogs is run from the vServer that hosts the Enterprise Controller component of the Exalogic Control stack, to collect diagnostics data for all the components on the rack, in serial mode.

  ```
  [root@rack-elcontrol bin]# sh ./exalogs --discover myrack.xml --all --serial
  ```

- In the following example, ExaLogs is run from a compute node, to collect diagnostics data for the Enterprise Controller vServer.

```
[root@el01cn01 bin]# sh ./exalogs --discover myrack.xml --emocec
```

> **Note:** Review the known issues listed in My Oracle Support document ID 1912063.1. For each issue, the document provides a description of the symptoms and causes, and a solution (if available).

4 Known Issues

See the My Oracle Support document ID 1912063.1.

5 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 customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Oracle Exalogic Elastic Cloud ExaLogs User's Guide, Release 1.3
E50008-03