Oracle Linux Automation Manager 2.2

Release Notes

G18170-04
March 2025

Oracle Linux Automation Manager 2.2 Release Notes,

G18170-04

Contents

7 Known Issues

Preface

Oracle Linux Automation Manager 2.2: Release Notes provides release information about Oracle Linux Automation Manager. This document includes information on component versions, new features, and documentation changes for Oracle Linux Automation Manager.

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.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at https://www.oracle.com/corporate/accessibility/.

Access to Oracle Support for Accessibility

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit https://www.oracle.com/corporate/accessibility/learning-support.html#support-tab.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

1 About Oracle Linux Automation Manager and Oracle Linux Automation Engine

Oracle Linux Automation Manager version 2.2, based on the open source projects Ansible and AWX, is a task engine and Web interface for scheduling and running Oracle Linux Automation Engine playbook tasks on the inventories the playbooks interact with. The Oracle Linux Automation Engine is an automation tool for deploying software, configuring systems, and orchestrating tasks such as upgrades and updates, in the form of playbooks.

Oracle Linux Automation Manager version 2.2 is based on the AWX version 23.7.0 open source software. The AWX development branch and documentation are maintained at https://github.com/ansible/awx/tree/23.7.0.

Oracle Linux Automation Manager, includes Oracle Linux Automation Engine which is based on the open source software package `ansible-core-2.16.6`. The development branch and documentation are maintained at https://github.com/ansible/ansible/tree/v2.16.6.

Ansible is a registered trademark of Red Hat, Inc. in the United States and other countries.

2 Component Versions

This section lists the version numbers of the major components included with Oracle Linux Automation Manager.

Table 2-1 Oracle Linux Automation Manager Components

NGINX	1.14
olam-ee:2.2 (ansible-core)	2.16
ol-automation-manager	2.2
ol-automation-manager-cli	2.2
postgresql	12 or 13 (13 on Private Automation Hub)
receptor	1.4
redis	5.0
Private Automation Hub (galaxy_ng)	4.9.1
Builder Utility (ansible_builder)	3.0.1

Table 2-2 Oracle Linux Automation Engine, Python Versions, OCI SDK, and Ansible OCI Collection

olam-ee Version	ansible-core Version	python Version	python-oci-sdk Version	oci-ansible-collection Version
2.2	2.16.6	3.11	2.137.1	5.3.0
2.2-minimal	2.16.6	3.11	n/a	n/a
2.1.2	2.15.3	3.11	2.85.0	n/a
2.1.1	2.15.3	3.11	2.85.0	n/a
2.1	2.12.2	3.8	2.85.0	n/a
2.0	2.12.2	3.8	2.85.0	n/a

3
New Features and Notable Changes

This section contains information on notable changes, release updates and new features. For more information about upgrading Oracle Linux Automation Manager, see Oracle Linux Automation Manager 2.2: Installation Guide.

Release 2.2

Some notable changes in Oracle Linux Automation Manager Release 2.2 are:

- Oracle Linux Automation Manager is now based on awx 23.7.0
- Builder Utility Python version is now 3.11
- `ansible-core` version is now 2.16.6
- Topology Viewer now available in the UI. For more information about using the viewer to verify the Oracle Linux Automation Manager server installation, see Oracle Linux Automation Manager 2.2: Installation Guide. This is a technology preview.
- The Oracle Linux Automation Manager images in Oracle Container Registry no longer use the `latest` tag. Always use the `2.2` tag when pulling images from the registry for the `2.2` release. The `latest` tag is deprecated and only applies to the Oracle Linux Automation Manager 2.1 release.

Release 2.1

Some notable changes in Oracle Linux Automation Manager Release 2.1 are:

- **Private Automation Hub**:
  This Oracle Linux Automation Manager feature is based on the galaxy_ng open source project that lets you synchronize custom collections and execution environment images to use with Oracle Linux Automation Manager deployments. Private Automation Hub can also synchronize collections and execution environments from remote container registries that you want to host locally. For more information about Private Automation Hub, see Oracle Linux Automation Manager 2.2: Private Automation Hub Installation Guide and Oracle Linux Automation Manager 2.2: Private Automation Hub User's Guide.
- **Builder Utility**
  The builder utility is based on the `ansible-builder` open source project that lets you customize and create execution environments and then upload them to Private Automation Hub. Being able to use customized container images as execution environments to run playbooks lets you ensure you have all the packages and dependencies you need on the container image necessary to run playbooks in a consistent and dependable way. For more information about the Builder utility, see Oracle Linux Automation Manager 2.2: Private Automation Hub Installation Guide and Oracle Linux Automation Manager 2.2: Private Automation Hub User's Guide.

Release 2.0

Some notable changes in Oracle Linux Automation Manager Release 2.0 are:

- **Service Mesh**: Service Mesh provides a multi-service network that links control and execution nodes within a secure mesh that enables the sharing of job execution. The Service Mesh can include up to 20 nodes. For more information about configuring the Service Mesh, see Oracle Linux Automation Manager 2.2: Installation Guide and Oracle Linux Automation Manager 2.2: User's Guide.
- **Control Plane**: The control plane is part of the Service Mesh that consists of control plane nodes that provide the user interface, role-based access control, and content management functionality. The Control Plane defines how automation is initiated, deployed, audited and delegated to the Execution Plane. From the Control Plane user interface or through the RESTful API, users can manage features such as inventory, schedule workflows, track changes, initiate reporting and so on. For more information, see Oracle Linux Automation Manager 2.2: Installation Guide and Oracle Linux Automation Manager 2.2: User's Guide.
- **Execution Plane**: The Execution Plane is part of the Service Mesh that consists of execution plane nodes that execute Oracle Linux Automation Engine playbooks. Execution plane nodes use a ready-built container with Oracle Linux, ansible-core, python and provides collections and libraries, which enables a consistent and defined environment every time they run. Execution environments replace python virtual environments. For more information, see Oracle Linux Automation Manager 2.2: Installation Guide and Oracle Linux Automation Manager 2.2: User's Guide.
- **Hop Nodes**: Hop nodes are connecting nodes that can link together cluster nodes within the Service Mesh, such as control and execution nodes, that cannot directly reach one another. These nodes do not appear as part of instance groups, but do appear as part of the Service Mesh peer relationships.
- **Remote Database Options**: You can now optionally install a PostgreSQL database on a separate host. For more information, see Oracle Linux Automation Manager 2.2: Installation Guide.
- **Upgrade Path from Release 1.0 to 2.0**: You can upgrade Oracle Linux Automation Manager Release 1.0 instances to Release 2.0. The upgrade path includes remaining on a single node instance to upgrading to a full clustered instance. For more information, see Oracle Linux Automation Manager 2.2: Installation Guide.
- **Workflow Templates**: You can create workflow templates using the Workflow Visualizer graphical tool. You can use the tool to specify the run sequence of disparate components such as job templates and management jobs, as nodes in a linear graph-like design. For more information, see Oracle Linux Automation Manager 2.2: User's Guide.
- **Instance Groups**: You can group control plane nodes and execution plane node into instance groups. By default, the Oracle Linux Automation Manager installation process creates a default instance group for control plane nodes and a default instance group for execution plane nodes. You can add or remove control and execution plane nodes to an instance group. And you can create additional instance groups for execution plane nodes to further manage what execution plane node runs a specific job. For more information, see Oracle Linux Automation Manager 2.2: Installation Guide and Oracle Linux Automation Manager 2.2: User's Guide.

Release 1.0.1

Some highlighted features in Oracle Linux Automation Manager Release 2.0 are:

- **Oracle Linux Automation Manager REST API**: You can now use the REST API to programmatically interact with Oracle Linux Automation Manager servers. The API is based on AWX version 15.0.1 open-source software and all upstream features are exposed in the REST API; however, support is limited to those features discussed in

Getting Started With Oracle Linux Automation Manager. For more information, see Oracle Linux Automation Manager 1.0: CLI and API Reference Guide.

- **Oracle Linux Automation Manager CLI**: You can now install and use the Oracle Linux Automation Manager CLI to interact with Oracle Linux Automation Manager servers. The CLI is based on AWX version 15.0.1 open-source software and all upstream features are exposed in the CLI; however, support is limited to those features discussed in Getting Started With Oracle Linux Automation Manager. For more information, see Oracle Linux Automation Manager 1.0: CLI and API Reference Guide.
- **Oracle Cloud Infrastructure Ansible Collection credential type**: Oracle Linux Automation Manager now includes the OCI credential type for accessing the OCI Ansible collection within an Oracle Linux Automation Engine playbook. If your Oracle Linux Automation Engine playbook uses the OCI Ansible collection, see https://docs.oracle.com/iaas/Content/API/SDKDocs/ansible.htm and find the setup instructions relating to AWX. The OCI credential type removes the need to manually create the OCI credential type as described in the Using Oracle Cloud Infrastructure with Ansible Tower and AWX blog post.

4
Documentation Changes

For the latest Oracle Linux Automation Manager Release 2.0 and Release 1.0 documentation, see Oracle Linux Automation Manager documentation.

Release 2.2

Release 2.2 includes the following notable changes to existing documentation:

- Oracle Linux Automation Manager 2.2: Installation Guide: Includes ugrade procedures to release 2.2.
- Oracle Linux Automation Manager 2.2: User's Guide: The uplift to awx 23.7.0 includes some UI changes in various locations.
- Oracle Linux Automation Manager 2.2: Private Automation Hub Installation Guide: Now includes ugrade procedures to release 2.2.
- Oracle Linux Automation Manager 2.2: Private Automation Hub User's Guide: Some changes to the Builder Utility discussions about Python versions and some small changes to the format 1 and format 2 examples.
- Oracle Linux Automation Manager 2.2: CLI and API Reference Guide: The uplift to awx 23.7.0 includes some changes to the REST API and CLI.

Release 2.1

Release 2.1 includes the following new documents:

- Oracle Linux Automation Manager 2.2: Private Automation Hub Installation Guide: This document provides instructions about installing, backing up, and restoring Private Automation Hub and installing the Builder utility.
- Oracle Linux Automation Manager 2.2: Private Automation Hub User's Guide: This document provides instructions about using Private Automation Hub to manage collections and execution environments for use with Oracle Linux Automation Manager. In addition, this document provides instructions for using the Builder utility to create custom execution environments and upload them to Private Automation Hub.

Release 2.1 also includes the following notable changes to existing documentation:

- Oracle Linux Automation Manager 2.2: User's Guide: A new section is available about creating execution environments for using custom execution environment container images hosted on Private Automation Hub or on some other local container registry. New instructions are available about creating credentials for accessing custom execution environments and about creating credentials for accessing collections hosted on Private Automation Hub.
- Oracle Linux Automation Manager 2.2: Installation Guide: Existing installation procedures now includes information about using custom execution environments when defining default execution environments when running playbooks in Oracle Linux Automation Manager.

Release 2.0

The contents of Oracle Linux Automation Manager 1.0: Getting Started has been split into the following books in release 2.0:

- Oracle Linux Automation Manager 2.2: Installation Guide: The Installation Guide provides the following information
  - Hardware requirements
  - Installation options
  - Service Mesh topology examples
  - instructions for installing on a single host with a collocated database
  - instructions for installing on a single host with a remote database
  - instructions for installing in a cluster of host with a remote database
  - Instructions for configuring the Service Mesh nodes
  - Instructions for adding and removing cluster nodes
  - Instructions for upgrading Oracle Linux Automation Manager release 1.0 to release 2.0
- Oracle Linux Automation Manager 2.2: User's Guide: The User's Guide provides information about setting up permissions, teams, and users, setting up resources, and using views. Notable additions in Release 2.0 include the following:
  - Setting up Work flow Templates
  - Creating Schedules for Resources
  - Viewing Execution Environments
  - Managing Instance Groups
- Oracle Linux Automation Manager 2.2: Administrator's Guide: The Administrator's Guide includes information about general administrative tasks, configuring credential types, configuring notification templates, scheduling management jobs, and configuring settings. Notable additions in Release 2.0 include instructions for setting up LDAP authentication for user accounts configured in an LDAP server that log on to Oracle Linux Automation Manager.

5 About the Oracle Linux Automation Manager Life Cycle

Support for product enhancements, Common Vulnerabilities, Exposures (CVEs) and bug fix updates are available for Oracle Linux Automation Manager as described in Oracle Linux: Product Life Cycle Information.

6 Obtaining Errata and CVE Notices

To be notified when Oracle releases new errata packages for Oracle Linux Automation Manager, you can subscribe to the Oracle Linux errata mailing lists at https://oss.oracle.com/mailman/listinfo/el-errata.

If you're logged in to ULN, you can also subscribe to these mailing lists by following the Subscribe to Enterprise Linux Errata mailing list links that are provided in the Errata tab.

Oracle publishes a complete list of errata made available on ULN at https://linux.oracle.com/errata. You can also see a published listing of Common Vulnerabilities and Exposures (CVEs) and explore their details and status at https://linux.oracle.com/cve. You can also track updates to Oracle Linux yum server repositories by visiting https://yum.oracle.com/whatsnew.html, where you can see which packages were updated within each repository for the previous six months.

7
Known Issues

This chapter contains information about known issues and limitations in this release.

Builder Utility Format 3 Unsupported

The Builder Utility can't build using format 3. Use format 1 or 2 instead.

Fail to Delete Namespace

A namespace can't be deleted if a collection is first uploaded to the namespace, then denied approval placing the collection in the rejected repository. If you try to delete the collection, the following error message appears:

```
Namespace "<namespace_name>" could not be deleted.
Error 400 - Bad Request: The server was unable to complete your request
```

In the previous example, *<namespace_name>* can be the name of any namespace. This error persists even if you delete the colletion in the rejected repository.

Workaround: Do the following:

1. Log in to the Private Automation Hub server.
2. Log in as the pulp user.

   ```
   su -l pulp -s /bin/bash
   ```

3. Run the following command:

   ```
   pulpcore-manager shell
   >>> from galaxy_ng.app.models.namespace import Namespace
   >>> Namespace.objects.filter(name="<namespace_name>").delete()
   (2,
   {'galaxy.CollectionImport': 1, 'galaxy.Namespace': 1})
   ```

   For example, the following command deletes the `oracle` namespace.

   ```
   pulpcore-manager shell
   >>> from galaxy_ng.app.models.namespace import Namespace
   >>> Namespace.objects.filter(name="oracle").delete()
   (2,
   {'galaxy.CollectionImport': 1, 'galaxy.Namespace': 1})
   ```

Job Status Change After Node Restart

When an execution plane node restarts while a job is running on the node, the control plane loses the node and reports the following job status:

```
Job reaped due to instance shutdown
```

In some cases, when the execution plane node is recovered, the failed job status description may change to the following:

```
JSON Failed to JSON parse a line from worker stream. Error: Expecting value: line 1
column 1 (char 0)
Line with invalid JSON data: b''
```

This occurs because the control node tries to get a status message from the failed job and this is the error message that's returned. You can ignore the new status for the failed job. Consider it as failed and restart if needed.

Container Error After Execution Node Restarts

If an execution plane node restarts while a job is running, the next job run on the node after it recovers could display an error in the stdout output for the playbook similar to the following:

```
ERRO[0000] Refreshing container
a16a37e423495ba0f5f10644617bf9b0ac874a8aae5ff9cdd43f3269fc3f1ac6: retrieving
temporary directory for container
a16a37e423495ba0f5f10644617bf9b0ac874a8aae5ff9cdd43f3269fc3f1ac6: no such
container
```

This error occurs because after the node restarts, Podman cannot find data for the directory of containers that were running before the node restarted. You can ignore the error message because it appears only on the first job after the node restarts. Subsequent jobs will not have this error.

Topology Viewer Download Bundle Fails

The download bundle function on the topology viewer feature returns the following error message:

```
"A server error has occurred."
```

This issue is being investigated.