Oracle Linux Automation Manager 2.1

Administrator's Guide

F80483-02
November 2024

Oracle Linux Automation Manager 2.1 Administrator's Guide,

F80483-02

Contents

Preface

Oracle Linux Automation Manager 2.1: Administrator's Guide describes administration tasks 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 Administering Oracle Linux Automation Manager

Oracle Linux Automation Manager provides features that allow you to manage your Oracle Linux installation. Using Oracle Linux Automation Manager you can:

- Perform general administrative tasks
- Configure new credential types
- Configure notification templates
- Schedule management jobs
- Configure general settings, such as LDAP authentication integration

2

About General Administrative Tasks

The following sections provide information about basic administrative tasks.

Starting, Stopping, and Restarting Oracle Linux Automation Manager

To start Oracle Linux Automation Manager , do the following:

1. Open a console.
2. Start the Oracle Linux Automation Manager service:

```
sudo systemctl start ol-automation-manager
```

To stop Oracle Linux Automation Manager , do the following:

1. Open a console.
2. Stop the Oracle Linux Automation Manager service:

```
sudo systemctl stop ol-automation-manager
```

To restart Oracle Linux Automation Manager , do the following:

1. Open a console.
2. Restart the Oracle Linux Automation Manager service:

```
sudo systemctl restart ol-automation-manager
```

Accessing Log Files

Oracle Linux Automation Manager and other applications generates log files that can be useful for troubleshooting various issues.

To access and review the log files, do the following:

1. Open a terminal on the system that is running Oracle Linux Automation Manager.
2. Go to the following locations to review application log files: .
   - For Oracle Linux Automation Manager log files, see `/var/log/tower`.
   - For NGINX log files, see `/var/log/nginx`.
   - For Redis log files, see `/var/log/redis`.
   - For Postgresql, see `/var/lib/pgsql/data/log` and `/var/lib/pgsql/initdb_postgresql.log`.

> **Note:**
>
> If the database is on a remote host, log into the host running the database to find the postgresql log files.

3. Review the logs.

> **Tip:**
>
> To find all error messages in all log files in the folder, use the following command: `cat * | grep -i error`.

Accessing Application Status

You can view Oracle Linux Automation Manager and other application statuses using the `systemctl` command that can be useful for troubleshooting various issues.

To view application status messages relating to Oracle Linux Automation Manager and other applications, do the following:

1. Open a terminal on the system that is running Oracle Linux Automation Manager.
2. Run the following commands:
   - For Oracle Linux Automation Manager, type `sudo systemctl status ol-automation-manager`.
   - For NGINX, type `sudo systemctl status nginx`.
   - For Postgresql, type `sudo systemctl status postgresql`.

     > **Note:**
     >
     > If the database is on a remote host, log into the host running the database to find the postgresql log files.

   - For Redis, type `sudo systemctl status redis`.
3. Review the application status messages and ensure that all are in the `Active (running)` state. Investigate any error messages.
4. If you find error messages from the status commands, you can investigate further by running the following commands:
   - For Oracle Linux Automation Manager, type `sudo journalctl -u ol-automation-manager`.
   - For NGINX, type `sudo journalctl -u nginx`.
   - For Postgresql, type `sudo journalctl -u postgresql`.

> **Note:**
>
> If the database is on a remote host, log into the host running the database to find the postgresql log files.

- For Redis, type `sudo journalctl -u redis`.

Backing Up and Restoring the Database

To back up the database, do the following:

1. Log in to the database.

   ```
   sudo su - postgres
   ```

2. Create a database dump file.

   ```
   pg_dumpall > olamv2upg.dump
   ```

3. Exit the database session.

   ```
   exit
   ```

To restore the database, do the following:

1. Log in to the database.

   ```
   sudo su - postgres
   ```

2. Create a database dump file.

   ```
   psql -d postgres -f /dirwithbackup/olamv2upg.dump
   ```

3. Exit the database session.

   ```
   exit
   ```

3
Configuring New Credential Types

Administrators can create new credential types using YAML/JSON. For more information about creating new credential types, see the upstream documentation.

To create a new credential type, do the following:

1. Log into Oracle Linux Automation Manager.
2. Expand the navigation menu, and from the Administration section, click **Credential Types**.

   The Credential Types page appears.
3. Click the **Plus** icon.

   The New Credential Type page appears.
4. In the Name field, enter the name for the new credential type.
5. In the Input Configuration field, enter input using JSON or YAML syntax.
6. In the Injector Configuration field, enter injectors using JSON or YAML syntax.
7. Click **Save**.

   You can now use the credential type when creating new credentials.

4
Configuring Notification Templates

You can create a notification template to define notification types and methods that can be generated at various times and for various resources.

To configure a notification template, do the following:

1. Log into Oracle Linux Automation Manager.
2. Expand the navigation menu, and from the Administration section, click **Notifications**.

   The Notifications page appears.
3. Click the **Plus** icon.

   The New Notification Template page appears.
4. In the Name field, enter the name for the new notification template.
5. From the Organization list, select an organization.
6. From the Type list, select a notification type, for example Email.
7. Review the upstream documentation for more information about configuring these notification types.
8. Click **Save**.

   You can now use this notification template with various resources.

5
Scheduling Management Jobs

Management Jobs are built-in Jobs relating managing aging data in Oracle Linux Automation Manager and include the following jobs:

- **Cleanup Activity Stream**: This management job removes activity stream history older than a specified age.
- **Cleanup Expired OAuth 2 Tokens**: This management job removes OAuth 2 access tokens and refresh tokens that are expired.
- **Cleanup Expired Sessions**: This management job removes browser sessions from the database that are expired.
- **Cleanup Job Details**: This management job removes job history older than a specified age.

To launch a management job, do the following:

1. Log into Oracle Linux Automation Manager.
2. Expand the navigation menu, and from the Administration section, click **Management Jobs**.

   The Management Jobs page appears.
3. Click the rocket icon.

   A dialog box may appear in some management jobs to specify how many days of data should be retained. If so, specify the number of days that data should be retained.
4. Click **Launch**.

To schedule a management job, do the following:

1. Log into Oracle Linux Automation Manager.
2. Expand the navigation menu, and from the Administration section, click **Management Jobs**.

   The Management Jobs page appears.
3. Click the calendar icon for the management job you want schedule.

   The Schedules page appears.
4. Click the **Plus** icon.

   The Add Schedule page appears.
5. In the Name field, enter the name of the schedule.
6. In the Start Date field, enter a start date.
7. In the Start Time field, enter the hours, minutes and seconds for the start time.
8. From the Local Time Zone list, select a time zone.
9. From the Repeat Frequency, select one of the following:
   - **None (run once)**
   - **Minute**

- **Hour**
- **Day**
- **Week**
- **Month**
- **Year**

When you select any option other than **None**, the Frequency Details area and fields appears.

10. In the Days of Data to Keep field, enter the number of days to retain data before deleting it.
11. In the Every field, enter the frequency to repeat the management job.
    - If you selected a Repeat Frequency of Minute, enter the number of minutes.
    - If you selected a Repeat Frequency of Hour, enter the number of hours.
    - If you entered a Repeat Frequency of Day, enter the number of days.
    - If you entered a Repeat Frequency of Week, enter the number of weeks. Additionally, the On Days list appears. Select one or more days of the week.
    - If you entered a Repeat Frequency of Month, do the following: Additionally, the On Day field appears. Enter the day of the month.
        a. Enter the number of months.
        b. If you want to specify a day on which start the management job, select **On Day**, and enter the day.
        c. If you want to specify a week and day, select **On The**, select the **first**, **second**, **third**, **fourth**, or **last week** of the month, and select the day of the week.
12. From the End list, select one of the following:
    - **Never**
    - **After**
    - **On Date**
13. Review the Schedule Description area dates.
14. Click **Save**.

6
Configuring Settings

You can configure general Oracle Linux Automation Manager settings from the Administration section on the Settings page. These settings are grouped into the following categories:

- **Authentication**: Provides general settings relating to authentication.
- **Jobs**: Provides general settings relating to jobs.
- **Systems**: Provides general settings relating to the Oracle Linux Automation Manager system.
- **User Interface**: Provides general settings relating to the Oracle Linux Automation Manager user interface.

For more information about these fields, see the upstream documentation.

Configuring LDAP Authentication

Administrators can integrate Oracle Linux Automation Manager's authentication mechanism with one or more existing Lightweight Directory Access Protocol (LDAP) servers for centralized user management and better integration with existing identity management platforms, such as Active Directory.

To configure LDAP authentication:

1. Log into Oracle Linux Automation Manager.
2. Expand the navigation menu, and click **Settings**.

   The Settings page appears.
3. In the Authentication panel on the Settings page, click **LDAP Settings**.
   The LDAP Settings page is displayed with multiple tabs to allow you to enter configuration details for several LDAP servers. Configure the Default LDAP server by clicking the **Edit** button and then enter information into the fields displayed on the Default tab:

   LDAP Server URI
   Provide the URI to access your LDAP server in the format: ldap://<*host*>:<*port*> where <*host*> is the host name of the LDAP server and <*port*> is the TCP port number that the LDAP server uses. This field is required. For example,

   ```
   ldap://ldap1.example.com:389
   ```

   If your LDAP server uses SSL, you can specify **ldaps** as the protocol within the scheme component of the URI. For example,

   ```
   ldaps://ldap1.example.com:636
   ```

   If your server uses StartTLS functionality, you can set the protocol to **ldap** within the URI scheme and enable the LDAP Start TLS option.

LDAP Bind DN
Provide the Distinguished Name (DN) used to authenticate Oracle Linux Automation Manager against the LDAP server using the Bind operation. This field is required if your LDAP server does not allow anonymous access. For example:

```
uid=admin,cn=users,cn=accounts,dc=example,dc=com
```

LDAP Bind Password
Provide the Bind password for the Bind DN that you provided above. Note that the password is encrypted within the Oracle Linux Automation Manager database and is not displayed as you type it.

LDAP Start TLS
Either enable or disable Start TLS encryption for your LDAP server, depending on whether it is configured to support this function and valid SSL/TLS certificates are properly configured on the server. Note that you must not enable this option if you have set the protocol to **ldaps** within the URI Scheme component of the LDAP Server URI.

LDAP User DN Template
Optionally configure an LDAP User DN Template that can be used to automatically authenticate against a particular DN for a user when a user name is provided. The template can use the **%(user)s** variable to automatically fill in the user name. For example:

```
uid=%(user),ou=Users,dc=example,dc=com
```

Providing a specific User DN Template can help to improve performance when authenticating against an LDAP server because it avoids a full search for the user, however it is not required and may reduce the flexibility of the authentication process where users may be configured under multiple DNs.

LDAP Group Type
Select an appropriate LDAP Group Type from the drop-down selector. This option defines how the LDAP server determines group membership for users when attempting to authorize them. LDAP Group Types map onto the ObjectClasses that are defined for your groups and may vary depending on your LDAP server. The option that you select here controls the filter used in the queries that are made to determine whether a user belongs to the LDAP Require Group or LDAP Deny Group.

LDAP Require Group
Optionally configure an LDAP Require Group by providing the DN of the group to which the users must belong to be authenticated. This option prevents users from authenticating in Oracle Linux Automation Manager unless they belong to a specific group of users. For example, the following establishes a group called olamusers as being a required group:

```
cn=olamusers,cn=groups,cn=accounts,dc=example,dc=com
```

LDAP Deny Group
Optionally configure an LDAP Deny Group by providing the DN of the group to which the users must not belong to be authenticated. This option is the opposite of the LDAP Require Group and prevents users from authenticating in Oracle Linux Automation Manager if they belong to the group specified. For example, the following establishes a group called engineers as being a denied group:

```
cn=engineers,cn=groups,cn=accounts,dc=example,dc=com
```

LDAP User Search
The LDAP User Search field allows you to configure a search DN, scope and filter to be used when determining whether a user is authorized to authenticate in Oracle Linux Automation Manager. It is also used to populate information about LDAP authenticated users in Oracle Linux Automation Manager when they are viewed on the Users page under the Access section of the navigation menu. For example:

```
[
  "cn=users,cn=accounts,dc=example,dc=com",
  "SCOPE_SUBTREE",
  "(uid=%(user)s)"
]
```

LDAP Group Search
The LDAP Group Search field allows you to configure a search DN, scope and filter to be used when determining which groups a user belongs to. This search query is also used to process the LDAP Organization and Team mappings. For example:

```
[
  "cn=groups,cn=accounts,dc=example,dc=com",
  "SCOPE_SUBTREE",
  "(objectClass=posixgroup)"
]
```

LDAP User Attribute Map
The LDAP User Attribute Map allows you to map LDAP attributes for a user entry to Oracle Linux Automation Manager attributes that are used to populate information about the user within the UI. The Oracle Linux Automation Manager attributes that can be mapped follow:

- email
- first_name
- last_name

You can map these attributes to the attribute entries for a user within your LDAP server. For example, the following describes an attribute map:

```
{
  "first_name": "givenName",
  "last_name": "sn",
  "email": "mail"
}
```

LDAP Group Type Parameters
This setting controls the parameters used when performing a group lookup on the LDAP server. There are two possible settings here:

```
{
"member_attr": "member",
"name_attr": "cn"
}
```

Note that if you are using Active Directory the `member_attr` attribute must not be set and should be excluded from the configuration.

LDAP User Flags By Group
The LDAP User Flags by Group field allows you to map LDAP groups to different roles within Oracle Linux Automation Manager. LDAP users belonging to a particular group can

be configured as `superusers` and users belonging to an alternate group can be configured as `auditors`. All other users that are authenticated are given standard user permissions within Oracle Linux Automation Manager. For example:

```
{
  "is_superuser": [
    "cn=olamadmins,cn=groups,cn=accounts,dc=example,dc=com"
  ],
  "is_system_auditor": [
    "cn=olamauditors,cn=groups,cn=accounts,dc=example,dc=com"
  ]
}
```

LDAP Organization Map

To configure Organization mappings for Oracle Linux Automation Manager you need to provide the mappings between your LDAP groups and your Organizations in Oracle Linux Automation Manager. Organizations are presented as keys within the JSON formatted string that you provide within this field. For each organization mapping you provide keys for the group entries for users and administrators within the group. Each map can have the following keys and values:

- **admins**:
  - **None**: organization admins are not updated based on LDAP values.
  - **True**: all users in LDAP are automatically added as admins of the organization.
  - **False**: no LDAP users are added as admins of the organization.
  - A string or list of strings that specify the group DN(s) to query for group members that should be assigned the admin role within the organization.
- **remove_admins**:
  - **True**: users that are not a member of the admins groups are removed from the organization's admin users.
  - **False**: users that are members of the admins groups are added to the organization's admin users.
- **users**:
  - **None**: organization users are not updated based on LDAP values.
  - **True**: all users in LDAP are automatically added as users of the organization.
  - **False**: no LDAP users are added as users of the organization.
  - A string or list of strings that specify the group DN(s) to query for group members that should be assigned the user role within the organization.
- **remove_users**:
  - **True**: users that are not a member of the users groups are removed from the organization's users.
  - **False**: users that are members of the users groups are added to the organization's users.

For example, for the Organization named "LDAP Group 1" you may have an entry similar to:

```
{
    "LDAP Group 1": {
```

```
      "admins":"cn=olamadmins,cn=groups,cn=accounts,dc=example,dc=com",
      "remove_admins": true,
      "users": [
           "cn=olamusers,cn=groups,cn=accounts,dc=example,dc=com",
           "cn=support,cn=groups,cn=accounts,dc=example,dc=com"
               ]
    }
}
```

LDAP Team Map

LDAP Team Maps are similar to LDAP Organization Maps and the primary key in each entry maps onto the Team rather than the organizational unit. Key options and values for each entry include:

- **organization**: an organization as defined either in your Organization mappings or within Oracle Linux Automation Manager, itself. If no entry is provided or the organization does not exist, an organization is created automatically.
- **users**:
  - **None**: team members are not automatically updated from LDAP.
  - **True**: all users in LDAP are automatically added as team members.
  - **False**: no LDAP users are added as team members.
  - A string or list of strings that specify the group DN(s) to query for group members that should be added as team members.
- **remove**:
  - **True**: users that are not a member of the users groups are removed from the team.
  - **False**: users that are members of the users groups are added to the team.

For example, the following team mappings may be created:

```
{
"LDAP Team 1": {
  "organization": "LDAP Group 1",
  "users": "cn=olamusers,cn=groups,cn=accounts,dc=example,dc=com",
  "remove": false
},
"LDAP Support": {
  "organization": "LDAP Group 1",
  "users": "cn=support,cn=groups,cn=accounts,dc=example,dc=com",
  "remove": true
}
}
```