Migrating from IBM Maximo Asset Management to IBM Maximo Manage

Contents

About this guide

This guide contains all information you need to migrate from Maximo® Asset Management to Maximo Manage as part of Maximo Application Suite. Use the information in Installing Maximo Application Suite as a checklist to ensure that you complete installation steps sequentially.

Note: Content is updated regularly, to ensure that you use the latest version, print the guide when you are ready to install.

Maximo Manage in Maximo Application Suite overview

IBM® Maximo Asset Management is now called Maximo Manage. Maximo Manage is one of many applications that are included within Maximo Application Suite.

Maximo Application Suite is an integrated suite of applications that is built on Red Hat® OpenShift® environment to provide multi-cloud portability, including support for Hybrid Cloud or on-premises deployments.

Maximo Application Suite also contains other applications such as IBM Maximo Health, IBM Maximo Predict, and IBM Maximo Monitor.

Maximo Application Suite is based on the Red Hat OpenShift deployment model. Red Hat OpenShift is a platform-as-a-service system that is built around containers and uses container orchestration that is provided by Kubernetes. It is designed to integrate well with IBM Cloud Pak® for Data components.

After you upgrade to Maximo Manage, user licenses are managed in Maximo Application Suite.

- In IBM Maximo Application Suite, subscription term or perpetual license is managed through AppPoints.
  - Contact IBM account team about AppPoints and IBM Maximo Application Suite license file.
  - Buy a pool of AppPoints.
  - The same AppPoints can be applied to any application in IBM Maximo Application Suite.

Available user entitlements:

- Limited
- Base
- Premium
- Self-Service

What's changed in Maximo Manage

In previous versions, Maximo Asset Management was an independent product. Now, Maximo Manage is part of Maximo Application Suite. Many processes such as licensing and user management that were managed at the product-level are now managed at the suite level.

Changes in Maximo Manage

Changes in licensing model

Maximo Application Suite uses a different licensing model and uses AppPoints to track application usage, runtime, and user access. AppPoints are allocated in your organization as defined by your license entitlement. You can configure your environment to enforce the AppPoint entitlement.

Changes in deployment and architecture

Installation, configuration, deployment, and upgrade are done by the Red Hat OpenShift operator in Maximo Application Suite. The entire deployment is based on Red Hat OpenShift.

Changes due to containerization

Containerization is the packaging of software code with just the operating system (OS) libraries and dependencies that are required to run the code to create a single lightweight executable package that is infrastructure-agnostic.System properties, server bundles, integration, and customization process when you upgrade from Maximo Asset Management to Maximo Manage are different because of containerization.

System properties

The Maximo properties file in Maximo Asset Management is replaced by the System properties application in Maximo Manage, through which you can set system properties in the Maximo

Manage user interface. System properties include global properties that apply to all the server instances that use a common database, including a clustered environment.

- The bootstrap properties, such as database username and password and encryption keys, are set in IBM Maximo Application Suite during deployment or in a custom resource in Red Hat OpenShift. These properties are applied to all workload deployments.
- The bundle-level properties for workloads are applied to the specific server and set in IBM Maximo Application Suite and stored as a ConfigMap file in Red Hat OpenShift. ConfigMap files contain configuration data that the Red Hat OpenShift pods consume.
- Use the System Properties application in Maximo Manage to set other properties. These properties are applied to all the workloads. For example, if you change the administrative password, you must update the **`mxe.adminPasswd`** property.

Server bundles
The Maximo Manage application can be deployed in one or more workloads called server bundles. A server bundle isolates the workload processes so that they can be independently managed. Server bundles can be independently scaled and managed based on your needs.

Integration
An integration framework integrates data with other applications, within your enterprise or with external systems. The framework includes predefined content and a toolkit to extend the predefined content to new integration points. It also enables message providers and an abstraction of message queuing features making Maximo Manage independent of messaging models like JMS or Kafka.

Customization process
Maximo Archiving is not supported in Maximo Application Suite. Maximo Manage uses customization archives for using customized Java™ classes or database scripts. Maximo Asset Management 7.6.1.2 included customization archives. However, Maximo Manage has enhanced the customization process further.

Changes in technology

RMI replaced by REST API
You use REST API instead of Remote Method Invocation (RMI) for interactions with the product from custom extensions or external applications. Maximo Application Suite does not support RMI.

Message queues
If you are using Service Integration (SI) buses, you must migrate to Apache Kafka or any other supported JMS provider.

Changes in application server
You do not need to install or migrate IBM WebSphere® Application Server unless you used any of its features outside Maximo Asset Management. IBM WebSphere Application Server Liberty is embedded in the Maximo image and deployed automatically by Maximo Manage. You can configure other server configurations in the Maximo Application Suite user interface. For more information, see Configuring the application server.

Changes in authentication and user management
In Maximo Asset Management 7.6, user authentication is configured in the application itself or at the application server level. For example, WebSphere Application Server Network Deployment.

In Maximo Manage, user authentication is configured at the Maximo Application Suite level. The upgrade process migrates the existing users to the **Users** application in Maximo Application Suite where you can view and edit their details. The users are synchronized from Maximo Application Suite to Maximo Manage by using a cron task. Security groups are configured in Maximo Manage.

Impact of upgrading from Maximo Asset Management to Maximo Manage

Upgrading from Maximo Asset Management to Maximo Manage has an impact on users and administrators.

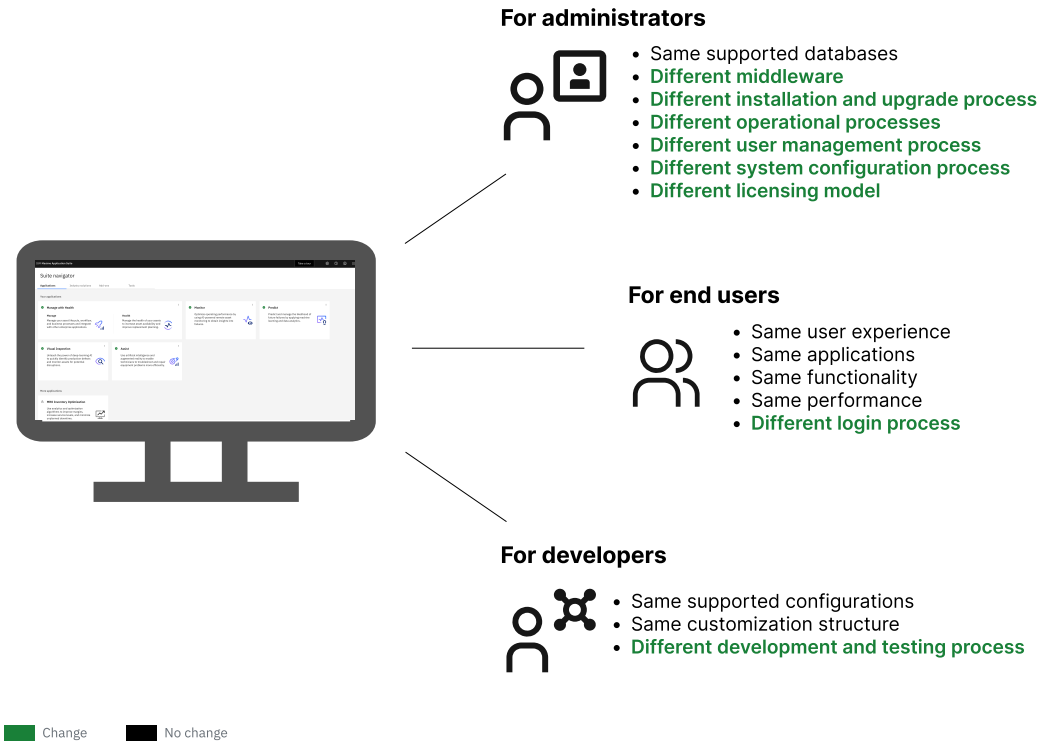


Figure 1. Impact of upgrading to Maximo Application Suite for different users, administrators, and developers

Changes for administrators

Maximo Manage supports the same databases as Maximo Asset Management. The following aspects are different when you upgrade to Maximo Application Suite:

Middleware implementation

Maximo Application Suite is deployed on Red Hat OpenShift Container Platform.

Installation and configuration

Maximo Application Suite installation and configuration process includes customer-managed or IBM managed installations. You can choose from multiple platforms and environments.

User management

User management process differs in that users are created and managed at the suite-level and are made available for application-level access with application-specific role assignments such as administrator or user. Users who are assigned a Maximo Manage entitlement, must be provided security group permissions in Maximo Manage to define the Maximo Manage applications, options, and data that the user can access.

System configuration

In Maximo Manage, the System Configuration module contains the Platform Configuration module and the Migration module. You use the applications in the Platform Configuration module to perform numerous tasks, such as managing systems properties and domains. You use the applications in the Migration module to migrate configuration content from one environment to another.

Upgrade process

You can upgrade Maximo Application Suite automatically or manually.

Changes for users

Maximo Application Suite users have a different login process.

Login process
The login process differs based on whether you have a customer-managed, IBM managed, or SaaS instance of Maximo Application Suite.

Changes for developers
If your role entails developing and testing Maximo Manage code locally, you are provided with more flexibility and can set up a local development environment.

What's supported in Maximo Application Suite

Products

IBM Maximo Anywhere and IBM Maximo Mobile
Maximo Anywhere is supported until Maximo Application Suite 8.8. Maximo Mobile has all connected and disconnected functions in one application. The initial version of Maximo Mobile includes Assist, Technician, and Inspections.

Industry solutions and add-ons
All industry solutions and add-ons that were a part of Maximo Asset Managementare supported in Maximo Application Suite.

Other products
IBM Maximo Scheduler, IBM Maximo Calibration and IBM Maximo Linear Asset Manager are now a part of Maximo Manage. IBM Maximo for Life Sciences is covered in Maximo Calibration.

SAP and Oracle integration
SAP and Oracle integrations supported in Maximo Manage through connectors.

Third-party add-ons
Third-party changes that exist in Maximo Asset Management are migrated to Maximo Manage only if the changes can be extracted completely in a customization archive.

Features

File system
For users that use the file system mount, the same file system mount can be configured in Maximo Application Suite through volume mount in Red Hat OpenShift. If the mount point is the same as in Maximo Asset Management, no change is needed for Maximo Manage configuration.

Object storage or S3
No change is needed to migrate attached documents. A public certificate for the object storage server might need to be imported.

Reporting
Business Intelligence and Reporting Tools (BIRT) supported.

Migration Manager
All pending migration packages must be migrated before the upgrade. Configuration options are provided during deployment for persistent volume mount for file-based packages. Maximo Anywhere APIs are available. Migration Manager automates creating and deploying packages.

Language support
All languages that were supported in Maximo Asset Management are supported in Maximo Manage.

What's not changed

Database
There are no significant changes in the database for Maximo Manage. Your Maximo Asset Management database is upgraded to the Maximo Manage database when you activate it.

User experience

- User interface and usability
- Applications
- Functions

- Performance

Architecture and deployment models

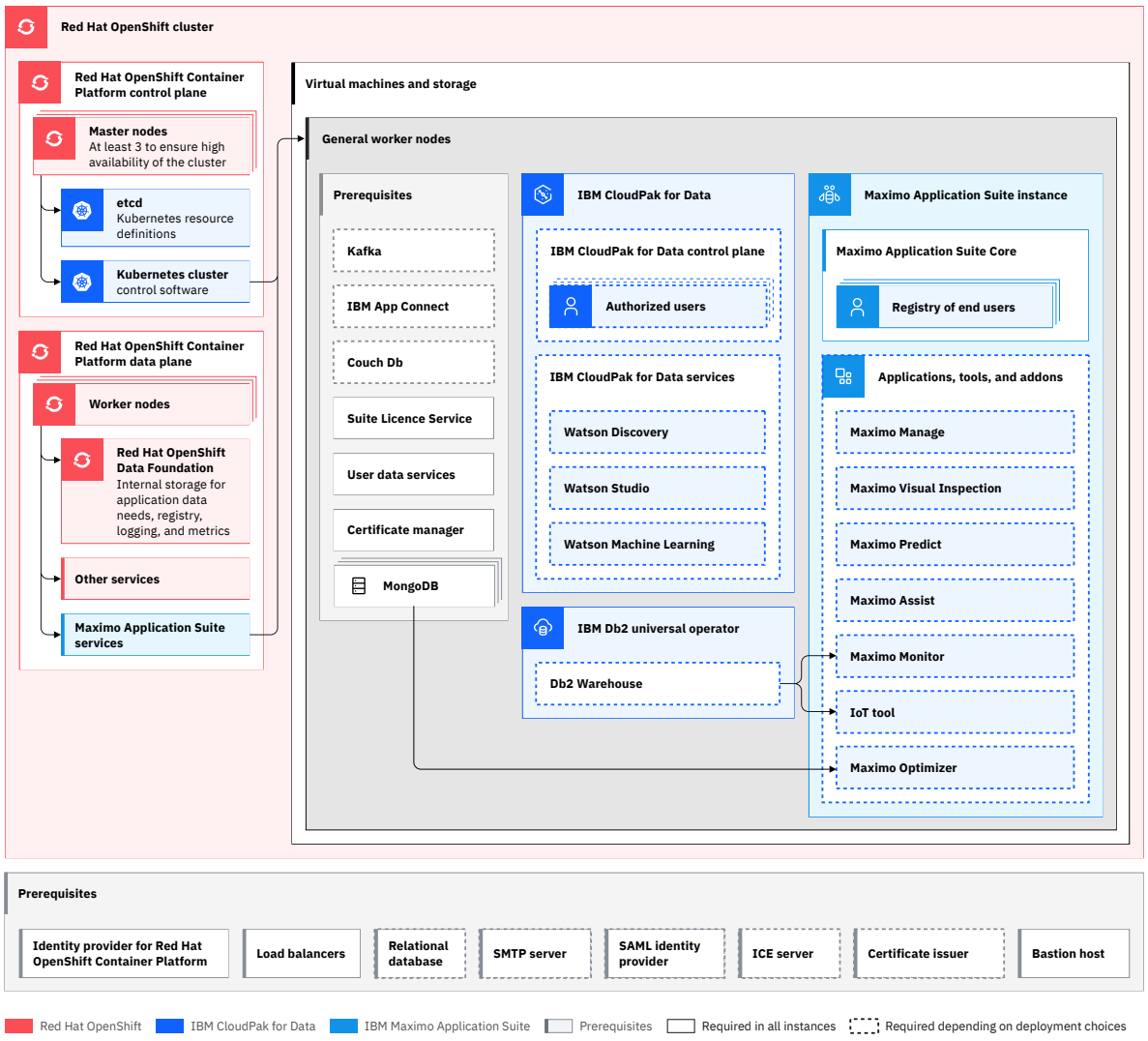
Maximo Application Suite architecture differs significantly from Maximo Asset Management because Maximo Application Suite is deployed on Red Hat OpenShift Container Platform. The shift in technology results in different deployment models for IBM Maximo Manage and Maximo Asset Management.

IBM Maximo Application Suite architecture

IBM Maximo Application Suite helps drive operational resiliency and reliability. With expanded access to enterprise asset management, Maximo Manage, Maximo Health, Maximo Predict, Maximo Monitor, and other applications, your team can reach across the enterprise to unify operations and maintain business continuity.

Maximo Application Suite has the following capabilities:

- Integrated suite of applications. For example, Maximo Manage, Maximo Health, Maximo Predict, and Maximo Monitor.
- Simplified licensing model.
- Hybrid-cloud deployment. Maximo Application Suite can be deployed on premises or in a public cloud.
- Comprehensive view of your assets.



- Maximo Application Suite includes applications, such as Maximo Manage, Maximo Monitor, Maximo Health, Maximo Predict, Maximo Visual Inspection, and Maximo Collaborate.
- IBM Cloud Pak for Data can run on your Red Hat OpenShift cluster. It offers AI-infused services for business and IT operations, development, data science, and management.

  Note: If you intend to integrate Maximo Manage with other application suites, you must install IBM Cloud Pak for Data. Otherwise, install only the IBM Db2® Warehouse stand-alone operator that can be one of the supported database options for Maximo Manage.
- MongoDB can exist in a cluster, cloud, or outside a cluster.
  - User, application, and entitlement metadata, such as OpenID Connect (OIDC) registration and user management, is stored in MongoDB. OIDC is an identity layer on the OAuth 2.0 protocol. OpenID Provider (OIDP) is an identity provider that is also used for user authentication.
- IBM Event Streams is a fully managed Apache Kafka-As-A-Service Platform for Cloud. The Maximo Manage application in the suite can be configured to use Kafka service.
- Maximo Manage support for all three databases, such as IBM Db2, Oracle Database, and Microsoft SQL Server.
- Maximo Manage can be configured to use cloud services, such as Cloud Object Storage, block, or file storage.

Deployment differences between Maximo Asset Management and Maximo Manage

Maximo Asset Management deployment is different from Maximo Manage deployment due to the change to Red Hat OpenShift Container Platform.

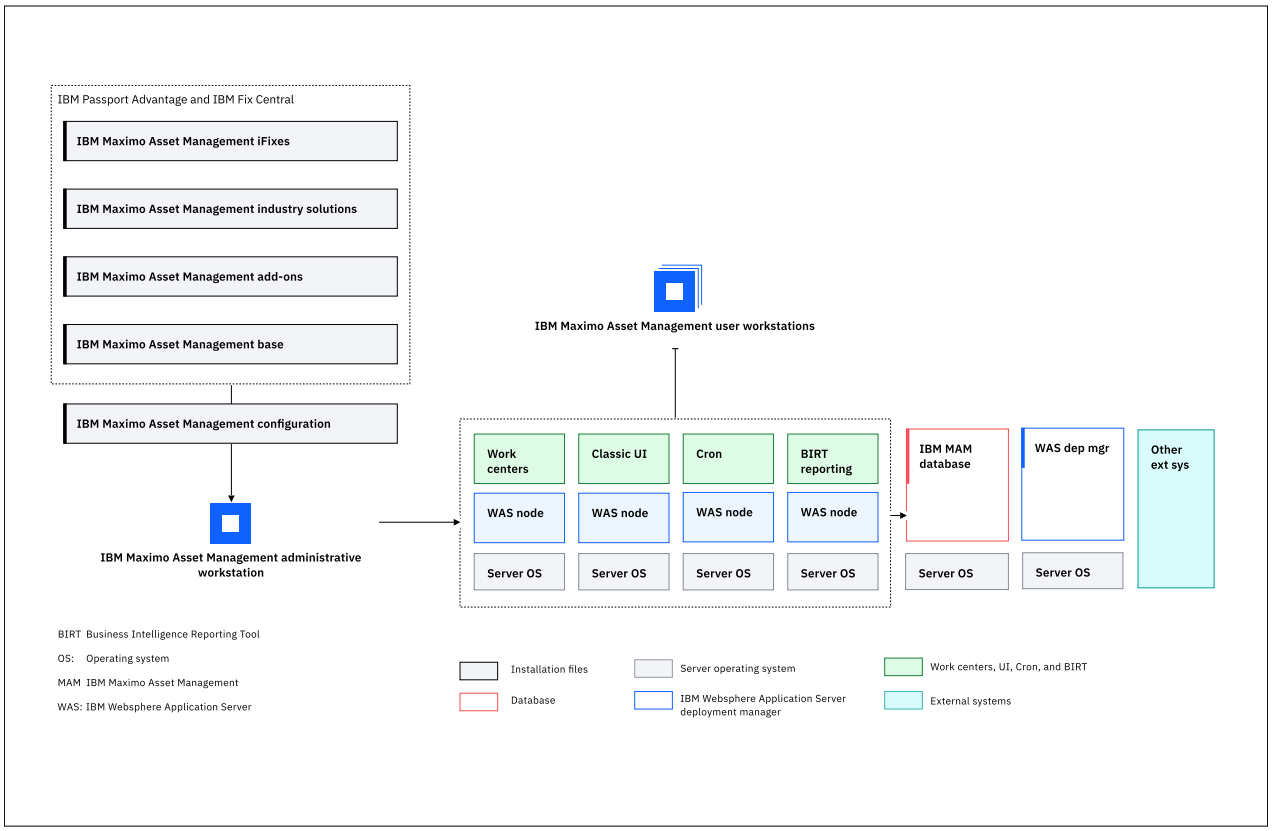


Figure 2. Maximo Asset Management 7.6 deployment architecture

Maximo Asset Management deployment

Maximo Asset Management 7.6 has an administrative workstation where the deployment software is downloaded. Deployment files are generated on the administrative workstation and deployed on an application server. The Maximo database and other external systems are run on separate servers.

In Maximo Asset Management deployment, the software is downloaded on the administrative workstation. The ear and war files are generated on the administrative workstation and deployed on an application server. In this deployment example, the cluster is running multiple servers for Classic UI, Work Centers, Cron, and BIRT reporting to distribute the load in a WebSphere Application Server Network Deployment. The Maximo Asset Management database and other external systems for integration are running on separate servers.

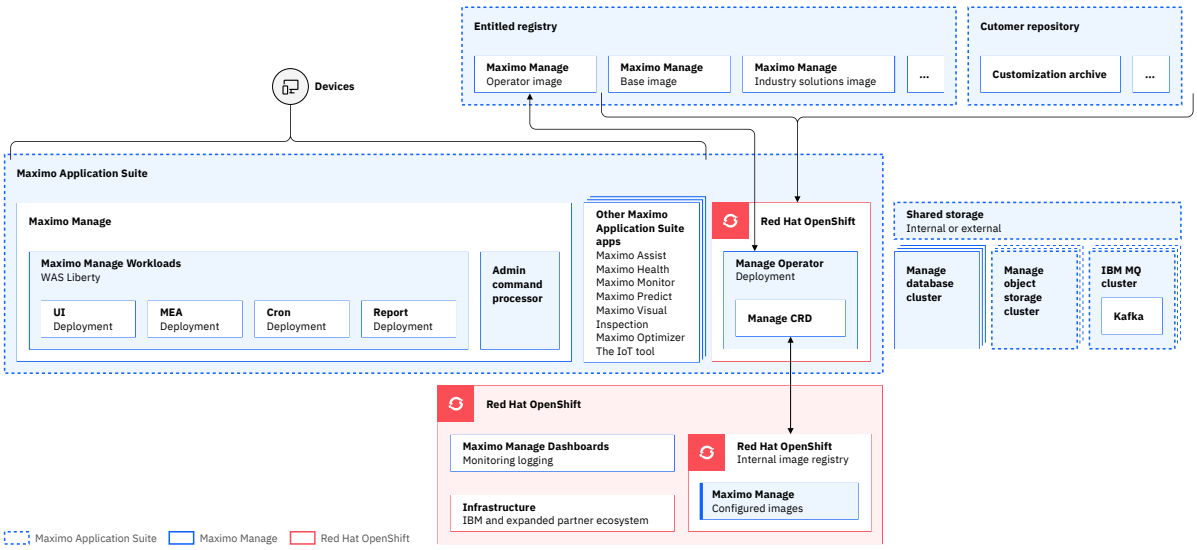
Maximo Manage deployment in Red Hat OpenShift Container Platform

When you deploy Maximo Manage in Red Hat OpenShift Container Platform, the Maximo Manage operator pulls the images from the IBM Entitled Registry and, if any customization exists, pulls the customization archive from the customer repository. The operator builds an Maximo Manage administrative image and configures images or workloads, such as UI, Cron, BIRT reporting.

The configured images and workloads are deployed to the Maximo Application Suite containers.

The built images are stored in the Red Hat OpenShift internal image registry repository.

The following diagram shows an example of a Maximo Manage application deployment in Red Hat OpenShift Container Platform:



Red Hat OpenShift
Red Hat OpenShift contains the infrastructure layer, and an internal image registry to store Maximo Manage configured images and services. Maximo Manage dashboards are provided in Red Hat OpenShift. You can use the dashboards to review logs and monitor Red Hat OpenShift and other applications that are deployed in Maximo Application Suite.

Images
The images for the Maximo Manage operator, Maximo Manage base, industry solutions, and add-ons are provided by the Entitled Registry. These images contain software application classes, deployment descriptors, XML, and scripts.

Maximo Manage operator
The Maximo Manage operator is similar to an installer. The operator pulls the images from the Entitled Registry. This process is similar to downloading software from Passport Advantage® and Fix Central.

Maximo Manage workloads or server bundles
Maximo Manage supports All, UI, Cron, Report, and Maximo Enterprise Adapter workloads. You can configure the workloads during the application deployment. In the diagram, the UI, Cron, Report, and Maximo Enterprise Adapter workloads are deployed.

Customization archive
The customization archive is stored in the customer repository and can be accessed from Maximo Application Suite by using HTTPS.

Admin command processor
The admin command processor pod runs Maximo Manage tools, such as the integrity checker.

Other Maximo Application Suite applications
You can deploy other Maximo Application Suite applications, such as Maximo Collaborate, Maximo Health, Maximo Predict, and Maximo Monitor.

Maximo Manage database
You can deploy the Maximo Manage database, such as IBM Db2 Warehouse, in a cluster.

Maximo Manage object storage
You can configure IBM Cloud Object Storage for document storage. For example, you can configure doc-links to use IBM Cloud Object Storage for storing documents.

Maximo Manage IBM MQ or Kafka
You can configure a messaging provider, such as JMS, IBM MQ, or Kafka, in a cluster.

Deployment from operator

The following steps describe how Maximo Manage is deployed by using the operator:

1. Create a custom resource (CR).

   An administrator selects, configures, and deploys the Maximo Manage application. The deployment creates a CR. A CR contains a user-entered configuration for the application, for example, the name, version, number of pods, database type and connection, type of workloads, and location of customization archive.

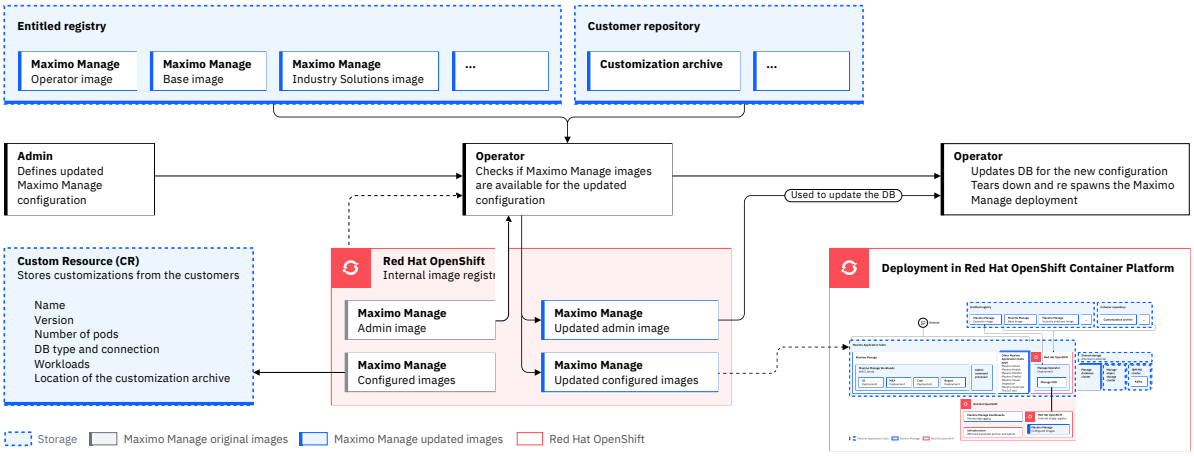
2. Create the image.

   The Maximo Manage operator pulls the images from the Entitled Registry and, if any customization exists, pulls the customization archive from the customer repository. The operator deploys the industry solutions or add-on images and the customization archive over the Maximo Manage base image to create the final images. The operator also validates for the dependency matrix. The operator creates an Maximo Manage administrative image and the Maximo Manage configured images or workloads. The final images are stored in the image registry repository.

3. Update the configuration.

   If the Maximo Manage database does not exist, the operator installs the Maximo Manage administrator image. If the database does exist, the operator upgrades the Maximo Manage database for a new configuration and restarts the Maximo Manage deployment. Maximo Manage configured images or workloads are deployed to containers in Maximo Application Suite.

The following diagram shows how the operator is used to deploy Maximo Manage.



Maximo Manage deployment model

In Maximo Manage deployment is done in Red Hat OpenShift clusters.

The following diagram shows a sample Maximo Manage deployment model.

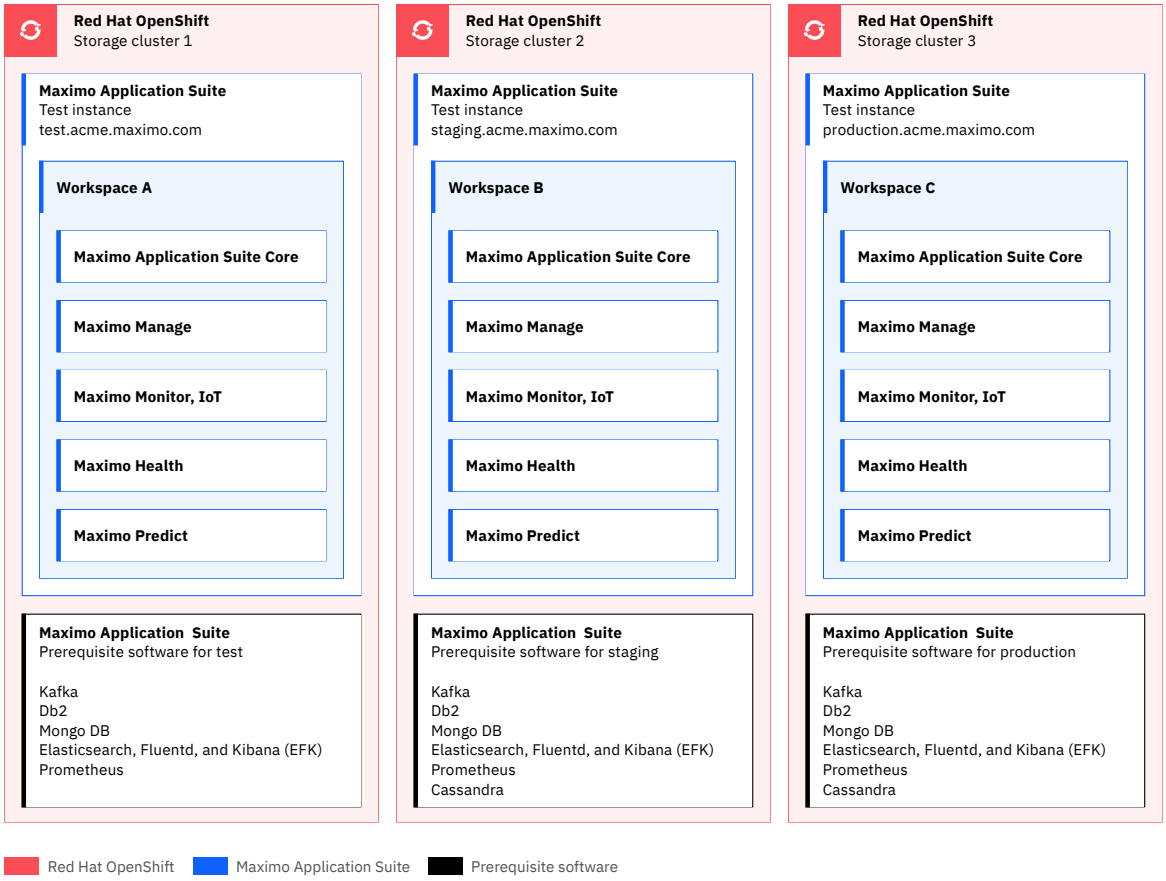


Figure 3. Maximo Manage deployment model sample

- Each cluster can run multiple Maximo Application Suite instances.
- Each instance runs its own set of pods and runtime code.
- All instances within the cluster can share a license pool.
- Each instance has its own workloads.
- Worker nodes capacity management is done per instance.
- Prerequisites stack is defined at the instance level.
- Each instance has an Red Hat OpenShift Data Foundation (ODF) storage cluster, that provides the necessary storage classes for all applications in Maximo Application Suite.

Deployment options for Maximo Application Suite

One of the main changes when you upgrade from Maximo Asset Management to Maximo Application Suite is the migration to Red Hat OpenShift Container Platform. Maximo Application Suite allows multiple ways to install and configure it.

All the applications, tools, add-ons, and utilities in Maximo Application Suite are based on three technology tiers.

Figure 4. Maximo Application Suite

- Tier 1 for data and analytics
  - IBM Cloud Pak for Data is a data and Artificial Intelligence (AI) platform with a data fabric that makes all data available for AI and analytics.
  - IBM Watson® Studio enables data scientists, developers, and analysts to build, run, and manage AI models, and optimize decisions on IBM Cloud Pak for Data.
  - IBM Watson Machine Learning provides tools and services to build, train, and deploy Machine Learning models.
- Tier 2 for a common operating environment uses Red Hat OpenShift Container Platform, which implements containerization by using Red Hat OpenShift clusters.
- Tier 3 is a cloud-based deployment to select deployment options best suited for your organization.

Note: The deployment options in the table are limited to options supported by IBM.			
Deployment	**Procurement**	**Provisioning and operation**	**Benefits**
On-premises - customer-managed	Customers buy Maximo Application Suite and use their own infrastructure.	Customers provision, manage, and operate the full technology stack.	Maximum operational flexibility
On-premises - hybrid-managed	Customers procure and manage infrastructure and application and avails PaaS services from IBM	IBM manages PaaS	PaaS services managed on both on-premises and hyperscalers

Note: The deployment options in the table are limited to options supported by IBM. *(continued)*			
Deployment	**Procurement**	**Provisioning and operation**	**Benefits**
Hyperscalers – customer-managed IBM or Amazon Web Services or Microsoft Azure	Customers buy software from IBM and infrastructure from hyperscalers.	Customers run IBM-provided automation scripts to deploy Maximo Application Suite on hyperscalers' cloud.	• Simplifies procurement and deployment • Allows customers to select their hyperscalers. • Flexibility for customers to manage and operate their environment.
	Customers buy software and infrastructure from hyperscalers.	Customers manage and operate both software and infrastructure.	
SaaS – IBM-managed	Customers buy a single part that includes software, infrastructure, and operations from either standard IBM sales channels or Amazon Web Services marketplace.	IBM provisions, manages, and operates customers' Maximo Application Suite environment on Amazon Web Services cloud by using IBM's cloud account.	• Reduced time-to-value • Reduced operational costs • Allows customers to focus on their business priorities.

Related information

Getting started with Maximo Application Suite as a Service

Planning your upgrade schedule

Upgrading involves planning, preparing, executing multiple activities and tasks, and troubleshooting. Planning enough time for all necessary activities helps ease the process.

Plan enough time to complete each phase of the upgrade process:

- Determining when you can upgrade
- Planning the upgrade
- Performing pre-upgrade tasks. For more information, see Before you upgrade.
- Upgrading in a test environment
- Troubleshooting your test upgrade
- Upgrading your production environment

Plan time into your schedule to perform the upgrade in a test environment. You can perform a test upgrade to test and troubleshoot your upgrade to avoid additional downtime in your production environment.

You can also schedule sufficient time to train administrators and users to use Maximo Application Suite and Maximo Manage. Team members need to understand the capabilities of the new software to participate in the upgrade planning process.

Training and courses

Maximo Application Suite is based on a different technology platform than Maximo Asset Management. IBM training can help you to upgrade your skills to manage the upgrade process, administer, and operate Maximo Application Suite and Maximo Manage.

Recommended skills for Maximo Manage administrators

Maximo Manage administrators perform the following tasks:

- Install and configure software.
- Understand Maximo Application Suite tasks, such as creating and modifying records.
- Understand relational database concepts, such as views and joins.
- Understand the Maximo Application Suite database and data relationships
- Construct Structured Query Language (SQL) statements.
- Understand the SQL syntax for your database.
- Set Maximo Application Suite properties for proper configuration.
- Define security privileges for users and groups.

Recommended skills for Maximo Manage users and developers

Maximo Manage users, depending on their access and entitlements could be customizing applications or using workflow for business processes. Users might benefit from training in the following areas:

- Customizing existing applications.
- If you are making changes or improvements to your business processes, training on the new processes.

References

To learn about	See
Red Hat OpenShift Container Platform	**Certification** • Red Hat Certified System Administrator (RHCSA) certification path for new learners • Red Hat Certified Specialist in OpenShift Administration certification path for experienced learners • Containerization and RHOCP essentials for Maximo Application Suite and Sterling solutions for an introduction on containerization and Red Hat OpenShift Container Platform essentials • Maximo Application Suite & Red Hat OpenShift Container Platform Deployment Technical Essentials for information on Maximo Application Suite and Red Hat OpenShift Container Platform installation. • Maximo Application Suite deployment overview for an overview on deploying Maximo Application Suite with an interactive flow chart and a series of demonstrations.
Maximo Mobile	Self-paced training Getting started with Maximo Mobile v2 to learn about a mobile solution that keeps technicians connected and your organization productive.

To learn about	See
Maximo Manage	Self-paced training • Maximo Application Suite - Manage: Introduction for an overview of the core functional areas of Maximo Manage and an introduction to the application relationships and overall usage. • Maximo Application Suite - Manage: Core Data Setup to learn how to create a new organization and site in Maximo Manage. • Maximo Application Suite - Manage: Users and Security to learn how to create new users in Maximo Manage and manage labor, crafts, and calendar records.
IBM Cloud®	If you have an Red Hat OpenShift deployment on IBM Cloud , Amazon Web Services, Microsoft Azure, or Google Cloud, you can deploy IBM Cloud Pak for Data on your cluster. You can also run Cloud Pak for Data on your private, on-premises cluster. For more information, see Overview of IBM Cloud Pak for Data.
Automate Maximo Application Suite installation	It is possible to automate some of the manual steps of installing Maximo Application Suite and its components, using Ansible collection roles.
MongoDB	Maximo Application Suite uses MongoDB for its data dictionary and local user management. Your MongoDB instance can run in the Red Hat OpenShift cluster or external to it. .
IBM Suite License Service	IBM Suite License Service provides features for managing virtualized environments and measuring license utilization. Suite License Service discovers the software that is installed in your infrastructure, helps you to analyze the consumption data, and generates audit reports. Each report provides you with different information about your infrastructure, for example the computer groups, software installations, and the content of your software catalog. For more information, see Suite License Service.
IBM Data Reporter Operator	An operator that accepts events and transforms them into reports that are submitted to the Data Service of the IBM Metrics Operator. For more information, see Data Reporter Operator.
IBM Cloud Pak for Business Automation	IBM Cloud Pak for Business Automation assembles certified software from the IBM Automation Platform for Digital Business on multiple cloud infrastructures. A private cloud vendor can be used as an enabling layer with a user interface and command line to limit access to members of an enterprise and partner networks. For more information, see Overview.

To learn about	See
Migrating from an existing Maximo Asset Management implementation	Migrating from an existing Maximo Asset Management implementation, from Maximo SaaS Flex or on premises, see Migration from SaaS Flex or On-Premise.
Selenium Automation Framework V3	Maximo Selenium Automation Framework V3 hosts test scripts for Maximo Asset Management 7.6.1.0 and earlier versions. For more information, see Maximo Selenium Automation Framework V3.  **Warning:** The Maximo Selenium automation framework tool is not officially supported by IBM.
IBM Maximo Test Automation Framework	As part of Maximo Application Suite, the IBM Maximo Test Automation Framework can be utilized to validate or re-validate Maximo Manage processes and capabilities for a given release based on approved and certified configurations from IBM. The Test Automation Framework consists of a series of validation test scripts that encompass asset and work management business processes. These test scripts contain information that can be built upon to develop and document a manufacturer's policies and procedures according to the implementation and use of Maximo Manage application software. These scripts can be used as quality assurance test cases to validate information systems. They are updated for every Maximo Application Suite long term supported release. For more information, see IBM Maximo Test Automation Framework.

Process overview

The migration process consists of an initial test deployment, which is followed by the production deployment after successful testing. Install and configure Maximo Application Suite, create a customization archive if you find it necessary, and finally deploy and activate Maximo Manage.

Note: If possible, deploy the Maximo Application Suite to multiple nonproduction environments for testing purpose before deploying the same on production environment.

The migration process supports the following elements of your Maximo Asset Management systems:

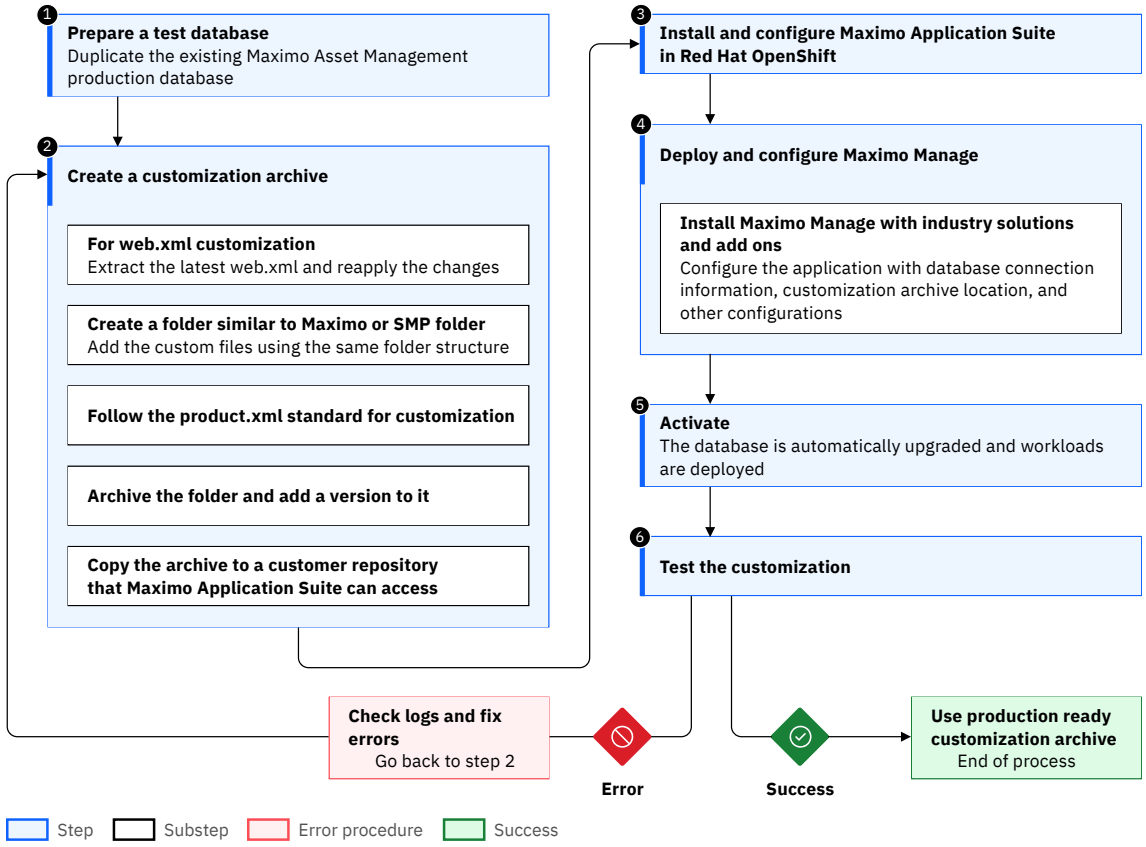
- All data
- All customizations. You must create a customization archive to store any specific changes, such as Java classes, XML files, and database scripts. You create the customization archive in a location accessible to IBM Maximo Application Suite during deployment. The structure of the customization archive is the same as the Maximo Asset Management folder structure.

  All customizations are preserved for Maximo Everyplace® during the upgrade process.
- Data model
- User interface and presentation layer
- Workflow processes
- Data validations and default values
- Escalations

When upgrading from Maximo Asset Management to Maximo Manage, the following items are not supported during the upgrade:

- The migration process does not support upgrading directly from Maximo Asset Management to Maximo Manage. You must install Maximo Application Suite first before you deploy Maximo Manage as an application within it.
- The migration process does not support migration of integration definitions specified in Maximo Asset Management. You must configure the integrations as part of the upgrade.
- Upgrading from one database platform to another. For example, you cannot upgrade from a Maximo Asset Management deployment that uses an Oracle database to a Maximo Manage deployment that uses a Db2 database.

The following diagram shows the migration process flow.



1 Prepare a test database
Duplicate the existing Maximo Asset Management production database to use as a test database.

2 Create a customization archive
For more information, see Customization archive guidelines.

3 Install and configure Maximo Application Suite
Install and configure a Maximo Application Suite instance in your Red Hat OpenShift environment. For more information, see Installing Maximo Application Suite.

Update all required properties and integrations after you install Maximo Application Suite. For more information, see “Updating system settings path” on page 118 and “Integrating with external systems” on page 108.

4 Deploy and configure Maximo Manage
Use the Maximo Application Suite user interface to configure Maximo Manage, industry solutions, and add-ons to use the new upgraded database, and other configurations. Specify the location of the customization archive. Maximo Manage application.

5 Activate Maximo Manage
Activate the Maximo Manage application. Activation updates the database and deploys workloads to the containers. For more information, see Activating Maximo Manage.

6 Test customization
Log on to Maximo Application Suite and launch Maximo Manage. Execute any business workflow or go to any page to check if the customization was migrated from Maximo Asset Management to Maximo Manage. For more information on a specific customization scenario, see Validating customization archive.

Planning to upgrade

All Maximo Manage implementations are unique, and the migration process is different for every deployment. However, some considerations in the process are common to every migration.

Upgrade checklist

You can use the upgrade checklist to check for tasks that you must do for upgrading from Maximo Asset Management to Maximo Manage.

Before you upgrade

- Ensure that you have Maximo Asset Management 7.6.0.10, 7.6.1.2, or 7.6.1.3 installed. For more information, see Installing Maximo Asset Management.

  Note: For IBM Control Desk, ensure that you have IBM Control Desk 7.6.1.5 and Maximo Asset Management 7.6.1.3 installed, to upgrade to Maximo Manage and Maximo IT.
- Plan for Maximo Application Suite installation requirements and preferences. For more information, see Planning.
- Install and configure Red Hat OpenShift cluster for non-production and production environments, according to your requirements.

  Note: To know more about installing on Amazon Web Services, IBM Cloud , Microsoft Azure, or using a command-line interface, see Supported installation paths.
- Become familiar with authentication, encryption and security, and SMTP configuration methods. For more information, see Authentication and security and SMTP configuration.
- Plan your upgrade schedule. For more information, see Planning your upgrade schedule.
- Review current database settings. For more information, see Database settings.
- If you are using Oracle Connector or SAP Connector, prepare it in Maximo Asset Management before you upgrade. For more information, see Preparing Oracle and SAP Connector before upgrade.
- Check for industry solutions and add-ons compatibility. For more information, see Deployment of industry solutions and add-ons.
- Disable custom triggers in any table of your database. For more information, see Custom triggers.
- Commit any database configuration changes that are pending. For more information, see Configuring the database.
- Backup your production database. For more information, see Backups.
- Prepare a test database as a duplicate of the Maximo Asset Management production database. For more information, see Backups.
- Complete any post-installation tasks for Maximo Asset Management before you upgrade. For more information, see Post installation tasks.
- Create a customization archive to store specific changes, such as Java classes, XML files, and database scripts. For more information, see Migrating customizations using customization archive.
  - Create deployment descriptors, as needed. A deployment descriptor describes how a component, application, or module is to be deployed with specific security settings, container options, and configuration requirements. For more information, see Deployment descriptors.
- Run Integrity checker in Maximo Asset Management 7.6.1.0, 7.6.1.2, or 7.6.1.3 and fix all errors reported. For more information, see Running Integrity checker.
- Test the upgrade in testing environment. For more information, see Testing the upgrade
- Use Maximo Manage logs to check and fix any errors you may run into while upgrading. For more information, see Troubleshooting the upgrade using Maximo Manage logs.

Tip: Keep your testing environment with application servers started, so that you can better measure the actual downtime it will take considering the production environment.

- Stop the application server only when you are nearing completion of Maximo Manage deployment as part of Maximo Application Suite.

During upgrade

- Install and configure Maximo Application Suite. For more information, see Installing Maximo Application Suite.
- Configure SMTP in Maximo Application Suite before you create the admin user. Otherwise, the email with the generated password cannot be sent. For more information, see SMTP server.
- Create and log on as the admin user in Maximo Application Suite. For more information, see Administering users and user access.
- Prepare Maximo Manage for deployment. For more information, see Preparing to upgrade.
- Deploy Maximo Manage. For more information, see Deploying in Maximo Application Suite.
- Set server bundle properties in Maximo Application Suite. For more information, see Adding server bundle properties.
  - Set `mxe.oslc.webappurl` to point to the route address for each server bundle, by using bundle level properties.
- Activate Maximo Manage. For more information see Activating Maximo Manage.
- Check Maximo Manage deployment status. For more information, see Checking Maximo Manage deployment status.
- Update system settings like `mxe.doclink.path01` and others with the new path. For more information, see "Updating system settings path" on page 118.
- If the database requires an SSL connection, you must obtain the certificate for the database. For more information, see Obtaining SS certificate for database.
- Import additional certificates as needed in Maximo Application Suite. For more information, see Importing additional certificates in Maximo Application Suite.
- Enable monitoring in Red Hat OpenShift cluster. For more information, see Installing Logging.
- Check Maximo Manage logs for any errors and fix them. For more information, see Troubleshooting using Maximo Manage logs.
- Check for user synchronization from Maximo Application Suite to Maximo Manage. For more information, see Managing users post upgrade.
- Configure Oracle Connector if you are using it. For more information, see Configuring Oracle Connector after upgrade.
- Configure SAP Connector, if you are using it. For more information, see Configuring SAP Connector after upgrade.
- **Tip:** The following points are optional as per your specific requirements.
- Maintain attached documents for your applications in persistent storage or cloud object storage, that is, S3. For more information, see Configuring attached documents.
- Use the External Systems application in Maximo Manage to initiate export and import of data, for example, integrate data by using files. For more information, see Exporting and importing file-based data.
- Use XSL maps to transform messages for outbound transactions in the integration framework provided with Maximo Manage. For more information, see XSL mapping.

Upgrade prerequisites

Check your Maximo Asset Management version and get your environment ready for upgrading by configuring your Red Hat OpenShift cluster.

Product version

Before you upgrade to Maximo Manage, you must have Maximo Asset Management 7.6.0.10 or 7.6.1.2 or later installed.

Note: For IBM Control Desk, ensure that you have IBM Control Desk 7.6.1.5 and Maximo Asset Management 7.6.1.3 installed, to upgrade to Maximo Manage and Maximo IT.

Red Hat OpenShift cluster requirements

Prepare the Red Hat OpenShift cluster based on the following requirements:

1. Determine the capacity needed to upgrade the product.

   For more information, see Planning.
2. Secure a Red Hat OpenShift cluster for development, testing, or production.

You must ensure that the prerequisites to install Maximo Application Suite are in place.

Requirements and capacity planning

Use the IBM Maximo Application Suite sizing calculator to estimate the required sizing for your planned deployment.

The Maximo Application Suite sizing calculator is used to estimate your Red Hat OpenShift worker node configuration requirements, storage requirements, and memory requirements.

1. Download the calculator.
   - Sizing calculator for 8.11 and earlier
   - Sizing calculator for 9.0
   - Sizing calculator for 9.0.1
2. Select or enter values for the yellow fields to match your planned application deployment.

   The calculator provides estimated total system requirements in VPCs and Memory (GB) for your configuration in the Resulting Complete Environments Requirements section of the Output table.

Important: The information in this document represents the minimum resources that you need to successfully install Maximo Application Suite. A minimum of 300GB of storage per worker node is recommended for Maximo Application Suite build process. You might need more resources to support your specific workload. If needed, work with your IBM Sales representative to generate more accurate calculations based on your expected workload.

For more information, see Sizing guidance.

Validating the upgrade process

Before you attempt the actual upgrade in a production environment, you could validate the upgrade process on multiple testing environments.

Procedure

1. In a test environment, check the logs for errors.
   a) If errors are found, fix them. If the problem is related to customization, use the admin image container in the Red Hat OpenShift environment to copy the entire build directory to a local development computer with the customization and compile.
   b) Create the customization archive again with the updated code, redeploy, and reactivate.
2. After successful testing, complete the upgrade by deploying in a production environment.
   a) Configure a Red Hat OpenShift cluster.
   b) Install Maximo Application Suite and all prerequisites.

c) Get all production configuration, database configuration, server bundles configurations, and customization archive, ready if they exist.

d) Deploy and activate Maximo Manage.

Preparing to upgrade

Before you upgrade, you must understand how authentication and security are implemented and how SMTP is configured. You prepare connectors and migrate customizations by using customization archives. Run the Integrity Checker utility to check for database errors before you start the upgrade process.

Reviewing database settings and backups

Before you start the upgrade, review your current database settings, disable custom triggers, commit any pending configuration changes, and backup the database and other important system files.

Review database settings

To ensure a successful upgrade, compare the configuration settings of your existing database with the default configuration settings of Maximo Manage. If your current values do not sufficiently match the default settings in Maximo Manage, it might cause problems during the upgrade process. Set configuration parameters that are equal to or greater than those parameters that required for Maximo Manage. For more information, see Preparing your database for deployment.

Disable custom triggers

Disable all custom triggers that exist for any table in your Maximo Asset Management database, for example, stored procedures, triggers, views, and synonyms. The upgrade process does not re-create or remove these objects. Reapply custom triggers after your database is upgraded as part of activating Maximo Manage.

Commit any database configuration changes

Commit any configuration changes to the Maximo Asset Management database before you upgrade to Maximo Application Suite. Configuration changes were part of the postinstallation tasks that were required when you installed Maximo Asset Management.

To confirm that all changes are committed, run the following SQL query against the Maximo Asset Management database:

```
SELECT count(*) from maxobjectcfg where changed != 'N'
SELECT count(*) from maxsysindexes where changed != 'N'
```

`'N'` indicates that a change is committed. If any positive row count values are returned for the query, you must apply or discard the configuration changes. Alternatively, you can use the **`configdb.bat`** command to commit configuration changes. For more information, see Configuring initial data.

Back up the existing database and other files

Back up the existing Maximo Asset Management database, the contents of the Maximo Asset Management installation folder, and the deployment engine. If a failure occurs during upgrade, you might be required to restore the Maximo Asset Management database. Some upgrade tasks cannot be rolled back after they are committed to the database. If you have a backup of the database, you can restore your environment. By default, files are found in the `C:\ibm\smp` directory. Backing up this directory can be useful if you must rebuild Maximo Asset Management EAR files. Back up the deployment engine registry as described in the Maximo Asset Managementinstallation information.

Related information

Authentication and security

Before you upgrade to Maximo Manage, you must be familiar with authentication, encryption and security, and SMTP configuration.

Authentication

- If you select local authentication, the username and password of users are stored on MongoDB and Maximo Application Suite directly authenticates the users.
- LDAP or SAML can be configured for authentication before or after the installation of Maximo Manage by using the **Configuration** page in Maximo Application Suite
  - Only LDAP registries that are supported by Liberty runtime can be used. For more information, see LDAP User Registry 3.0
  - If you are currently using LDAP or SAML, your existing configuration can be used in Maximo Application Suite.
  - From Maximo Application Suite 8.8 onwards, Maximo Manage supports only API key-based authentication for integration with external applications and REST API transactions. For integration, XML along with SOAP and HTTP protocols use API keys. The existing REST APIs, for example, **`maxrest`** or **`rest`** support API keys as well as new REST APIs added from Maximo Application Suite 8.8 like **`oslc`**.

    API key-based authentication is primarily used for machine to machine interactions and authentication. If you are using **`maxauth`** in Maximo Asset Management, after upgrading to Maximo Application Suite and deploying Maximo Manage, use API keys because **`maxauth`** is not supported in Maximo Manage.

Encryption and security

If you are using custom encryption keys for CRYPTO and CRYTOX attributes, then the custom keys must be provided during the Maximo Manage application installation.

Custom encryption properties are specified in the `maximo.properties` file in Maximo Asset Management.

Note: The `maximo.properties` file is not used in Maximo Application Suite.

After the Maximo Manage application deployment, if encryption keys are not specified when you activate Maximo Manage with a fresh database, new encryption keys are automatically generated. Set the **`autoGenerateEncryptionKeys`** property to false if you do not want to generate the keys automatically.

SMTP configuration

SMTP configuration is retained in the database during the upgrade.

It is used by Maximo Manage for sending emails.

Maximo Application Suite also provides SMTP configuration. This configuration is used for sending welcome emails and user password emails. For more information, see Configuring SMTP.

Preparing enterprise adapters before upgrade

If you are using IBM Maximo Enterprise Adapter for Oracle Applications or IBM Maximo Enterprise Adapter for SAP Applications, you must prepare the connector before you upgrade.

Procedure

1. Stop all transactions between Maximo Asset Management and IBM Maximo Enterprise Adapter for Oracle Applications or IBM Maximo Enterprise Adapter for SAP Applications.
2. Process all transactions in the interface tables and integration queues.

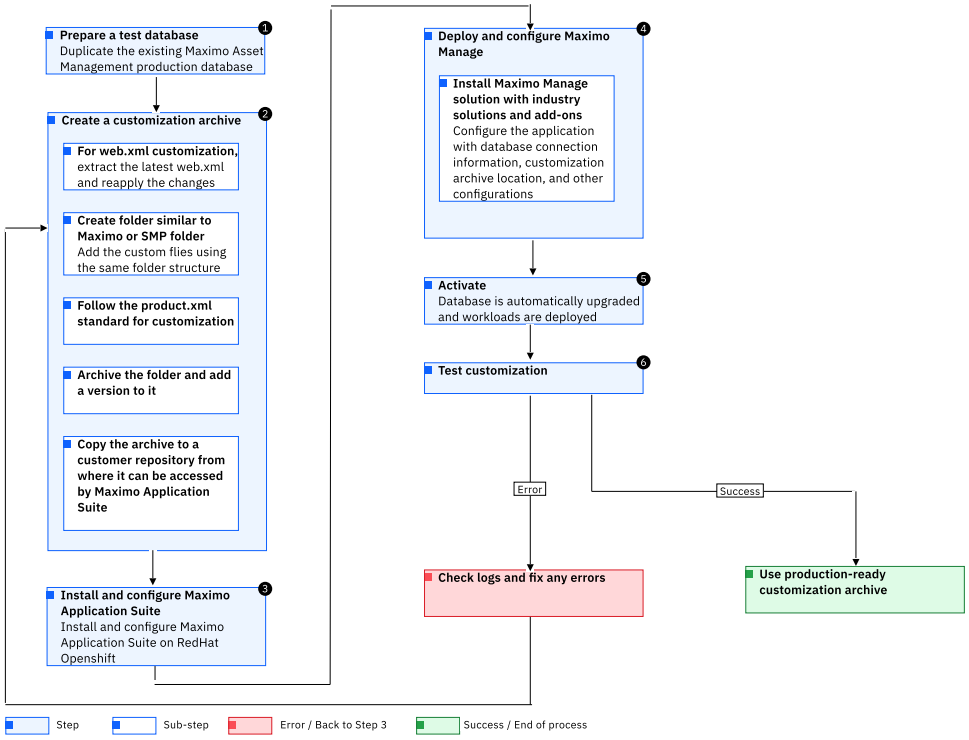
3. Resolve any Oracle integration errors in Maximo Asset Management. For more information, see Error management.
4. In the External Systems application, disable the external system for the enterprise adapter.
   - The OA12 external system for Oracle.
   - The SAP2005 external system for SAP Connector
5. If you are preparing an Oracle connector, back up PL/SQL user exits and any customization on the Oracle e-business suite.
6. Stop the Maximo server.
7. Back up the Maximo database.

Migrating customizations using customization archives

All customer-specific changes, such as Java classes, XML files, and database scripts, must be included in a customization archive. You create the customization archive in a location accessible to IBM Maximo Application Suite during deployment. The structure of the customization archive is the same as the Maximo Asset Management folder structure. Test the customization archive in a development or test environment before you apply it to the production environment.

About this task

The following diagram shows the customization process:



Procedure

1. Prepare the database.

   Prepare a test database as a duplicate of the existing Maximo Asset Management production database.

2. Create a customization archive.

   For more information, see Creating customization archives section. If your customization includes a `web.xml` file, such as customer servlet, filters, changes in the order of the servlet startup, context parameters, or session timeout:

   a) Install Maximo Manage with industry solutions and add-ons without customization on an empty database.

   b) Extract the `web.xml` file.
   Use the **`oc rsync`** command to retrieve the `web.xml` file.

   c) Apply your changes.

   d) Copy the `web.xml` file to the customization archive in the appropriate directory.

   Note: Ensure that the location of the customization archive is accessible by the Red Hat OpenShift.

3. Deploy the application.

   Use Maximo Application Suite to configure Maximo Manage, industry solutions and add-ons to point to the database to upgrade and other configurations. Specify the location of the customization archive. Deploy.

   For more information, see Setting up a local Maximo Manage development environment.

4. Activate the application.

   Maximo Manage updates the database and deploys workloads to the liberty containers.

5. Test the application.

6. Using the admin image pod that contains `maxinst`, in the Red Hat OpenShift console, copy the entire build directory with the customization and compile. You can use IDE to build the Maximo Manage project with customizations.

What to do next

After you fix any errors, create a customization archive again with the updated code. Deploy and activate the Maximo Manage application. After successful testing, use the customization archive in the production environment.

Creating customization archives

You can create a customization archive, which is a set of files that contain changes and customizations for the application.

Procedure

1. Follow the existing SMP or Maximo folder structure to create a customization folder.
2. Copy class files (Java customization) in the appropriate directory and extracted with the existing Maximo binary files in the correct package or module hierarchy.

   The file might include the following information.

   - Maximo Business Object (MBO) customization
   - Field validation
   - Maximo integration framework
   - Others like customer-specific code

What to do next

- Version your customization archive.
- Customization archive location is specified during Maximo Manage application configuration from IBM Maximo Application Suite admin dashboard.
- Customization archive location must be accessible from Maximo Application Suite. It is a zip file that you can access through HTTP, HTTPS, FTP, or FTPS.

- Any database scripts can be added to customer directory and an `a_customer.xml` file must be used. For more information, see Customization archive guidelines. All those files need to be part of the customization archive. Update db will run after deployment to apply all customer scripts.

Deployment descriptors

All deployment descriptors such as `web.xml`, `ejb-jar.xml`, and `webservices.xml` files that are customized by users can be put into the customization archive and overlaid on the files that are supplied by IBM.

- The following files need to be created:
  - For deployment of full Maximo Asset Management in one deployment (Maximo **-all**)
    - `deployment\was-liberty-default\config-deployment-descriptors\maximo-all\maximouiweb\webmodule\WEB-INF\web.xml`
    - `deployment\was-liberty-default\config-deployment-descriptors\maximo-all\maximo-x\webmodule\WEB-INF\web.xml`
    - `deployment\was-liberty-default\config-deployment-descriptors\maximo-all\maximo-x\webmodule\WEB-INF\web-guest.xml`
    - `deployment\was-liberty-default\config-deployment-descriptors\maximo-all\maximo-x\webmodule\WEB-INF\web.xml`
    - `deployment\was-liberty-default\config-deployment-descriptors\maximo-all\maxrestweb\webmodule\WEB-INF\web.xml`
    - `deployment\was-liberty-default\config-deployment-descriptors\maximo-all\maximo-x\webmodule\WEB-INF\web.xml`
    - `deployment\was-liberty-default\config-deployment-descriptors\maximo-all\mboweb\webmodule\WEB-INF\web.xml`
    - `deployment\was-liberty-default\config-deployment-descriptors\maximo-all\meaweb\webmodule\WEB-INF\web.xml`
    - `deployment\was-liberty-default\config-deployment-descriptors\maximo-all\mboejb\ejbmodule\META-INF\ejb-jar.xml`
  - UI deployment
    - `deployment\was-liberty-default\config-deployment-descriptors\maximo-ui\webmodule\WEB-INF\web.xml`
  - Maximo Enterprise Adapter
    - `deployment\was-liberty-default\config-deployment-descriptors\maximo-mea\meaweb\webmodule\WEB-INF\web.xml`
    - `deployment\was-liberty-default\config-deployment-descriptors\maximo-mea\mboejb\ejbmodule\META-INF\ejb-jar.xml`
  - Report deployment
    - `deployment\was-liberty-default\config-deployment-descriptors\maximo-report\webmodule\WEB-INF\web.xml`
  - Cron deployment
    - `deployment\was-liberty-default\config-deployment-descriptors\maximo-cron\webmodule\WEB-INF\web.xml`
- IBM-specific file `ibm-ejb-jar-bnd.xml` is not used in Liberty deployment.

XSL customization

XSL customizations that are part of the Maximo EAR or WAR files are also part of the customization archive and are copied to final images. For more information, see Rule-based customization.

Sample customization archive

A customization archive contains Java classes, XML files, scripts, servlet, and deployment descriptors.

A sample customization archive has the following elements.

- classes
  - applications\maximo\businessobjects\classes\cust\app\asset\Asset.class
  - applications\maximo\businessobjects\classes\cust\app\asset\AssetSet.class
  - applications\maximo\businessobjects\classes\cust\app\asset\FldAssetNewField.class
- product xml
  - applications\maximo\properties\product\a_customer.xml
- script
  - tools\maximo\en\cust\V7612_01.dbc
- servlets
  - applications\maximo\commonweb\classes\com\ibm\tivoli\maximo\oslc\provider\MyPingServlet.class
- deployment descriptors (web.xml)
  - deployment\was-liberty-default\config-deployment-descriptors\maximo-mea\meaweb\webmodule\WEB-INF\web.xml
  - deployment\was-liberty-default\config-deployment-descriptors\maximo-ui\meaui\webmodule\WEB-INF\web.xml

Adding third-party JAR files

You can add third-party Java Archive files to the Manage lib folder in the customization archive to migrate any extended third-party functionality added in Maximo Asset Management to Maximo Manage.

About this task

Use the following steps to add a third-party JAR file name to the Manifest file, if the deployment has an all or mea bundle server type.

Procedure

1. Go to the admin pod, which is the name of the pod that contains **maxinst**, terminal and get the file.
2. Go to /opt/IBM/SMP/maximo/deployment/was-liberty-default deployment folder.
   a) Get the maximo-all.xml file if the deployment is all bundle server type. In maximo-all.xml, go to the maximo.businessobjectclasspath property name and add the JAR file name in the path. When deployed it will update the classpath in the Manifest file.
   b) Get the buildmaximomea-ear.xml file if the deployment has a mea bundle server type and update the classpath.
3. Copy the updated file in the Customization Archive folder in the same path and archive or compress the folder.

   The file path for the **maximo-all.xML** file is <localdrive>\custasset_bin\deployment\was-liberty-default.
4. Deploy the customization archive and activate the changes.

   For more information, see Customizing the application.

Running Integrity Checker before upgrade

The Integrity Checker is a database configuration utility that you can use to assess the health of the base layer data dictionary.

About this task

You can use the Integrity Checker utility to ensure that the Maximo Asset Management database is ready for upgrade. When run in Report mode, the Integrity Checker utility checks the current database and reports errors. If the Integrity Checker reports an error, you must resolve it by running the Integrity Checker in Repair mode.

Procedure

1. On the system where Maximo Asset Management is installed, open a command prompt and change directory to the tools directory.
   For example, *install_home*\maximo\tools\maximo.
2. Start the Integrity Checker utility by issuing the **integrityui.bat** command.
3. Select the **Check Integrity** tab.
4. Run the Integrity Checker in Report mode.
   a) Ensure that the **Repair Mode** check box is cleared and then click **Run Integrity Checker**.
   b) When the report dialog box appears, click **OK**.
      Results are found in the install_home\maximo\tools\maximo\log directory in the file that is defined in the **Log File Name** field of the **Check Integrity** pane.
5. Optional: If any errors are reported, run the Integrity Checker in Repair mode.
   a) Select the **Repair Mode** check box and then click **Run Integrity Checker**.
   b) When the report dialog box appears, click **OK**.
   c) Change directory to install_home\maximo\tools\maximo\ and then run the **configdb** command.
      For more information on Integrity checker warning and error messages, see Integrity checker messages.

      Attention: Although the Integrity Checker can repair many issues, you might need to resolve some errors manually by consulting the log files or opening a case with the IBM support team.

What to do next

Check the log file to ensure that all reported items are repaired. If further manual intervention is required, you must resolve the errors and then rerun the Integrity Checker in Report mode. Repeat the process until no more errors are reported.

Global property values

Set the **mxe.int.globaldir** and other properties when you upgrade from Maximo Asset Management to Maximo Manage.

If you want directories to exist for the doc-link table and for integration, check that **mxe.int.globaldir** and other URL properties are set. Check whether you need to mount additional persistent volumes or change directories for transient data.

For more information, see Troubleshooting global property values.

Installing Maximo Application Suite

To install IBM Maximo Application Suite, plan your installation, select a supported installation path, and set up the initial configuration.

Supported installation paths

Use the Maximo Application Suite command line interface (CLI) for a standard installation on multiple platforms such as x86/amd64, IBM Z, and IBM Power. Use the Ansible® DevOps collection with CLI for advanced users to build installation topologies beyond what is possible with the Maximo Application Suite CLI.

Alternatively, customize the Maximo Application Suite installation with Amazon Web Services CloudFormation templates for Amazon Web Services user accounts and Microsoft Azure Resource Manager templates for Microsoft Azure user accounts.

The standard installation with CLI works on any Red Hat OpenShift Container Platform instance, such as Installer Provisioned Infrastructure (IPI), User Provisioned Infrastructure (UPI), Single Node Red Hat OpenShift, or IBM Cloud Kubernetes Service. The customized installations by using Amazon Web Services CloudFormation and Microsoft Azure Resource Manager templates work with Red Hat OpenShift Container Platform instances for existing, IPI, or UPI Red Hat OpenShift cluster.

For more information, see Detailed system requirements.

Standard installation with IBM Maximo Application Suite CLI

You can install the IBM Maximo Application Suite by using a command-line interface (CLI) utility.

Before you begin

You can use IBM's container image to install Maximo Application Suite. The container image provides an out of the box environment for managing Maximo Application Suite on Red Hat OpenShift, with numerous dependencies preinstalled.
In a scenario where you need to install the CLI utility, which is an open source tool, on your local system, you must configure the following software.

- Bash (v4)
- Red Hat OpenShift client
- IBM Cloud client with container plug-in enabled

  Note: IBM Cloud is not required if you are deploying Maximo Application Suite on an organization's Red Hat OpenShift cluster.
- Ansible
- Python
- Network access to the Red Hat OpenShift cluster

Download the Maximo Application Suite CLI utility onto the bastion host. This utility provides commands to manage the local docker registry, configure policies in the Red Hat OpenShift cluster, and deploy Maximo Application Suite.

For more information about installing the utility, see Maximo Application Suite CLI Utility.

To use the Maximo Application Suite CLI utility, ensure that the bastion host has support for running docker containers.

For more information, see Planning for IBM Maximo Application Suite standard installation with CLI.

Procedure

1. Run the **docker run** command to use the container image that is published.

   ```
   docker run -ti --rm -v ~:/mnt/home --pull always quay.io/ibmmas/cli
   ```

   If you want a specific release of the image, use a specific version tag in the **docker pull** command :

   ```
   docker run -ti -v ~:/mnt/home quay.io/ibmmas/cli:x.y.z
   ```

   For more information on the software included in the container image, see MAS CLI Base Image.
2. Install Maximo Application Suite.

   Run the following command from a running CLI docker container.

   ```
   docker run -ti --pull always quay.io/ibmmas/cli mas install
   ```

   For more information, see the Install command.

   Tip: To customize Maximo Application Suite instances with extra features that might be unavailable in Maximo Application Suite CLI, you can use ansible-devops for advanced configuration. For more information, see Maximo Application Suite Devops Ansible Collection.

   For a list of supported CLI commands, see CLI commands.

   You can also run the installation in interactive mode. For more information, see Interactive install.

What to do next

Complete the Maximo Application Suite setup. For more information, see the following topics.

Authentication methods
Maximo Application Suite supports MongoDB, Lightweight Directory Access Protocol (LDAP) authentication, and Security Assertion Markup Language (SAML) authentication methods for local user authentication.

LDAP user registry synchronization
User registry synchronization simplifies Maximo Application Suite user management by synchronizing users and groups between an LDAP server and your local Maximo Application Suite user registry.

Administering users and user access

The initial superuser account is used to complete the Maximo Application Suite setup. You can add application administrator users or system administrator users for day-to-day administrative tasks.

- Administering users and user access in Maximo Application Suite in 9.0 and earlier
- Administering users and user access in Maximo Application Suite in 9.1

Getting started
With the setup completed, your users can log in and start using Maximo Application Suite.

IBM Maximo Application Suite installation in disconnected environments

You can install IBM Maximo Application Suite in an air gap environment, which is also known as disconnected, offline, or restricted network.

Before you begin

Before you install Maximo Application Suite, ensure you download the necessary software and set up the environment.

You must have access to the following components:

- A private image registry setup and running in the restricted network, and secured with certificates. Configure one of the following options:

- – A bastion host with access to product images on the internet and the restricted network and has support for running docker containers. The docker image that contains the IBM Maximo Application Suite command line utility on the bastion host.
  – A host outside the restricted network with access to product images on the internet and with support for running docker containers. The docker image that contains the Maximo Application Suite command line utility on the host. Portable disk space sufficient to store the required images. A host inside the restricted network with support for running docker containers and can access the images downloaded to the portable disk space.
- Red Hat OpenShift cluster setup as an air gap cluster for disconnected installation.

  Ensure you use the Maximo Application Suite sizing calculator to estimate your Red Hat OpenShift Worker Node configuration, storage, and memory requirements.

  For more information, see Requirements and capacity planning.
- IBM entitlement key.
- IBM Maximo Application Suite license file.

For more information, see Prerequisites for installing.

Note: It is recommended that you follow the Red Hat OpenShift instructions for setting up a docker registry.

An alternative is deploying the docker registry into a separate Red Hat OpenShift cluster. This scenario requires two Red Hat OpenShift clusters. One Red Hat OpenShift cluster for the docker registry and a second air gap Red Hat OpenShift cluster for Maximo Application Suite.

You can run the following command to deploy a docker registry into an Red Hat OpenShift cluster.

```
docker run -ti --rm --pull always quay.io/ibmmas/cli mas setup-registry
```

Remember: You must not use the same Red Hat OpenShift cluster for both, the docker registry and Maximo Application Suite.

About this task

Run the following docker commands on a bastion host to install Maximo Application Suite.

Procedure

1. Select one of the static catalogs and automatic approval strategy for your installation.

   For more information, see Catalog selection.
2. Mirror the container images.

   You can use three modes to mirror the container images.

   - **`direct`** mirrors images directly from the source registry to your private registry.
   - **`to-filesystem`** mirrors images from the source to a local directory.
   - **`from-filesystem`** mirrors images from a local directory to your private registry.

   Run the following command to mirror the product images that is necessary to install and run Maximo Application Suite.

   ```
   docker run -ti --pull always quay.io/ibmmas/cli mas mirror-images
   ```

   Tip: You can also use this command to mirror the images for Red Hat OpenShift.

   The **`mirror-images`** command accepts the name of a static catalog to control what is mirrored to your registry. It mirrors the necessary images of the latest package version in that catalog in your private registry.

   You are prompted to set the target registry for mirroring the image to choose a catalog and the subset of content that you want to mirror. You can either mirror everything from the catalog or control exactly

what is mirrored to your private registry. Controlling what is mirrored reduces the registry storage requirements and the time and bandwidth that is used to mirror the images. For more information, see Catalog selection.

For example, run the **mirror-images** command non interactively to directly mirror images from the internet to your private registry using a bastion host.

```
mas mirror-images \
  -m direct \
  -H myprivateregistry.com -P 5000 -u $REGISTRY_USERNAME -p $REGISTRY_PASSWORD \
  -c v8-221025-amd64 --mirror-core --mirror-iot --mirror-optimizer --mirror-manage \
  --ibm-entitlement $IBM_ENTITLEMENT_KEY \
  --redhat-username $REDHAT_USERNAME --redhat-password $REDHAT_PASSWORD \
  --no-confirm
```

For more information, see Mirror images command.

3. Configure Red Hat OpenShift to use your Private Registry for Maximo Application Suite.

   Your cluster must be configured to use the private registry as a mirror for the Maximo Application Suite container images. An `ImageContentSourcePolicy` named `mas-and-dependencies` is created in the cluster, which is also the resource that the Maximo Application Suite install uses to detect whether the installation is a disconnected installation and tailors the options that are presented when you run the **mas install** command.

   Run the following command from a running CLI docker container to configure the Red Hat OpenShift instance to use your Maximo Application Suite private registry.

   ```
   docker run -ti --pull always quay.io/ibmmas/cli mas configure-airgap
   ```

   Provide information about the private registry, including the CA certificate necessary to configure your cluster to trust the private registry.

   You can run the command in noninteractive mode.

   ```
   mas configure-airgap \
     -H myprivateregistry.com -P 5000 -u $REGISTRY_USERNAME -p $REGISTRY_PASSWORD \
     --ca-file /mnt/local-mirror/registry-ca.crt \
     --no-confirm
   ```

   For more information, see Configure air gap command.

4. Install Maximo Application Suite.

   Run the following command from a running CLI docker container to install Maximo Application Suite.

   ```
   docker run -ti --pull always quay.io/ibmmas/cli mas install
   ```

   For more information, see the Install command.

What to do next

Perform the initial configuration tasks for setting up the Maximo Application Suite. For more information, see “Setting up IBM Maximo Application Suite” on page 91.

Starting in Maximo Application Suite 9.0, 8.11.7, 8.10.10 or later versions, you can also upload your usage data to IBM by using the IBM Data Reporter Operator . For more information, see Data Reporter Operator.

IBM Maximo Application Suite installation with Amazon Web Services CloudFormation templates

You can install IBM Maximo Application Suite in the Amazon Web Services (AWS) cloud by using the Amazon Web Services CloudFormation templates. Maximo Application Suite is available as a bring-your-own-license (BYOL) and contract pricing product in AWS Marketplace. After you configure the installation requirements and consider your installation preferences, you subscribe to the product, specify the installation parameters, and start the installation.

In your AWS account, the installation process creates the virtual network infrastructure and the Red Hat OpenShift cluster, and then installs the Maximo Application Suite prerequisites and Maximo Application Suite. If you configured a verified email address in the Amazon simple email service (SES), you receive emails that contain the information that you need to access Maximo Application Suite.

Installing the Maximo Application Suite on Amazon Web Services

To install IBM Maximo Application Suite on Amazon Web Services (AWS), you configure the prerequisite components, consider your installation preferences, specify the installation criteria in an AWS CloudFormation stack template, and create the stack.
You must configure prerequisites and gather the information that you need to specify the installation parameters. For more information, see Planning to install on Amazon Web Services.

In addition, you must consider your installation preferences, such as the type of Maximo Application Suite offering that you want and whether you want to create a Red Hat OpenShift cluster or reuse an existing cluster.

Note: The existing cluster must be created by using the automated deployment option.

For more information, see Installation considerations.

Maximo Application Suite on Amazon is available as a bring your own license (BYOL) and as a paid offering.

Installing BYOL IBM Maximo Application Suite

The IBM Maximo Application Suite (BYOL) can be installed from the Amazon Web Services Marketplace based on your infrastructure needs. You can install a new Red Hat OpenShift cluster before you install Maximo Application Suite or you can use your existing Red Hat OpenShift cluster to install Maximo Application Suite.

Before you begin

Before you can install Maximo Application Suite on Amazon Web Services, you must configure prerequisites and gather information that you need to complete the installation.

For more information, see Prerequisites for installing Maximo Application Suite on Amazon Web Services.

Consider other criteria, such as the type of Maximo Application Suite offering that you want and whether you want to create a Red Hat OpenShift cluster or reuse an existing one.

Note: The existing cluster must be created by using the automated deployment option only.

For more information, see Preparing to install Maximo Application Suite on Amazon Web Services.

This product requires an internet connection to deploy properly. The following code is downloaded on deployment for setting up the Maximo Application Suite Red Hat OpenShift Container Platform cluster:

- https://github.com/ibm-mas/ansible-devops.git
- https://github.com/ibm-mas/multicloud-bootstrap.git

About this task

The following three fulfillment options (CloudFormation templates) are available for installing Maximo Application Suite:

1. New Red Hat OpenShift cluster by using the Installer Provisioned Infrastructure (IPI)
2. New Red Hat OpenShift cluster by using the User Provisioned Infrastructure (UPI)
3. Existing Red Hat OpenShift cluster

Note: Starting in 8.11, for US GovCloud regions, you can install Maximo Application Suite in private hosted zones for existing Red Hat OpenShift cluster and New Red Hat OpenShift cluster by using the User Provisioned Infrastructure (UPI).

Procedure

1. In the AWS Marketplace service console, click **Discover product** and search for `IBM Maximo Application Suite (BYOL)`. The product is sold by IBM Maximo.
2. Open Maximo Application Suite.
3. Review the product information to select a fulfillment option and click **Continue to Subscribe**.
4. In the subscription page, to create the subscription, review the terms and conditions and click **Accept Terms**.
5. After the subscription is created, click **Continue to Configuration**.
6. In the configuration page, in the **Region** field, select a supported geographical region where you want to install the Maximo Application Suite.

   For the list of supported regions, see Amazon Web Services region for installing Maximo Application Suite.
7. Accept the default values in the other fields and click **Continue to Launch**.
8. In the **Launch** page, select **Choose Action** > **Launch CloudFormation**.
9. To open the CloudFormation stack wizard, click **Launch**.
10. In the **Create stack** wizard step, accept the default values, and click **Next**.
11. In the **Specify stack details** step, in the **Stack name** section, enter a unique name.
12. In the **Parameters** section, enter the installation parameters by using the information that you gathered when you configured the Prerequisites for installing Maximo Application Suite on Amazon Web Services and Preparing to install Maximo Application Suite on Amazon Web Services.
    a) Enter the mandatory parameters.
       - The Maximo Application Suite offering type, such as Maximo Application Suite Core and Cloud Pak for Data or Maximo Application Suite and Maximo Manage.

         Starting in 8.11, for US GovCloud regions, you can configure the offering type Maximo Application Suite Core or Maximo Application Suite and Maximo Manage. Cloud Pak for Data is not available for configuration.
       - All parameters in the **Cluster and bootnode access** and **Keys and licenses** sections.

    b) To reuse an existing Red Hat OpenShift cluster, complete the Existing Red Hat OpenShift cluster connection details section.

       Alternatively, to create a new cluster, complete the steps that are given in Connection details for an existing Red Hat OpenShift cluster section.

    c) To reuse an existing network infrastructure, complete the steps that are given in Connection details for using an existing network infrastructure section.

    d) In the optional parameter groups, such as the group of IBM Maximo Manage database configuration parameters, ensure that you either specify all parameter values or leave all empty.

       Note: Follow the instructions commented in the field to know how to fill them. For the field `MASManageDBJdbcUrl`, you can specify it by using one of the following JDBC URL formats. Ensure that the Port that is used contains the SSL enabled port of the database.

       IBM Db2

       For Db2 SSL database connections in Maximo Manage, you must specify `sslConnection=true`. Ensure that you use a semicolon to end the JDBC connection string. You can use the following URL as an example:

       ```
       jdbc:db2://mymaximodb.com:50001/MAXDB:sslConnection=true;
       ```

       Oracle Database

       Starting in 8.11, for US GovCloud regions, you can configure an Oracle Database. You must ensure that the Oracle Database follows the Federal Information Processing Standard (FIPS).

       For more information, see Configuring Oracle Database

Note: If you choose to install Maximo Application Suite with Manage, you can use the default IBM Db2 instance that is provisioned by IBM instead of configuring your own external Db2 instance. To configure the default IBM Db2, do not add information in the username, password, JDBC URL, certificate URL, and demo data fields.

The internal Db2 configuration is available from Maximo Application Suite 8.10 or later.

13. To configure Amazon Managed Streaming for Kafka, select **Yes**.

    The Amazon Managed Streaming for Kafka is configured to process streaming data of applications such as IoT and IBM Maximo Monitor from Maximo Application Suite.

    You can configure the default Amazon Managed Streaming from Maximo Application Suite 8.10.

    Starting in 8.11, for US GovCloud regions, Amazon Managed Streaming for Kafka configuration is not required.
14. To configure a DocumentDB instance, you can select any of the following options.
    - Configure a new MongoDB community edition.
    - Use an existing MongoDB

      Add existing MongoDB connection details such as the username, password, MongoDB hostname, and CA certificate.

      Starting in 8.11, for US GovCloud regions, you must ensure your existing MongoDB connection is FIPS compliant.
    - Configure a new Amazon DocumentDB.
    - Use an existing Amazon DocumentDB.

      Add existing DocumentDB connection details such as the username, password, DocumentDB hostname, and CA certificate.

      Add the VPC ID of the region in which the Amazon DocumentDB is deployed. Ensure that the DocumentDB does not have a matching or overlapping `IPv4 CIDR block 10.0.0.0/16`.

    You can configure the default DocumentDB from Maximo Application Suite 8.10.

    Starting in 8.11, for US GovCloud regions, DocumentDB is not available for configuration.
15. In the **Other parameters**, select the **OperationalMode** as `Production` or `Non-production`.

    You can specify deployments for Production or Nonproduction environments. Nonproduction installations can be used for internal development and testing. The installation AppPoints are unused in the Nonproduction installations. These specifications are also visible in the metrics that are shared with IBM and on the product UI.
16. Click **Next**.
17. In the **Configure stack options** step, configure any additional options that you require. To know more about the stack options, click the "Learn more" links available in each option. Click **Next** when done.
18. In the **Review** step, review the values that you entered and acknowledge the message that relates to identity and access management (IAM) resources.
19. To begin the installation, click **Create stack**.

What to do next

During the installation process, the AWS CloudFormation stack template that you configured is used to create a Bootnode. The Bootnode contains all required resources to complete the installation. To verify that the Bootnode is created successfully, in the **CloudFormation** > **Stacks** page, confirm that the stack status is updated to `CREATE_IN_PROGRESS`

For more information, see “Monitoring IBM Maximo Application Suite installation on Amazon Web Services” on page 48.

Installing client managed IBM Maximo Application Suite for public paid offer

The paid AWS product can be installed from the AWS Marketplace based on your infrastructure needs. A new Red Hat OpenShift instance is created as part of the Maximo Application Suite installation.

Before you begin

Before you can install Maximo Application Suite on Amazon Web Services, you must configure prerequisites and gather information that you need to complete the installation.

For more information, see Prerequisites for installing Maximo Application Suite on Amazon Web Services.

Consider other criteria, such as the type of Maximo Application Suite offering that you want and whether you want to create a Red Hat OpenShift cluster or reuse an existing one.

Note: The existing cluster must be created by using the automated deployment option only.

For more information, see Preparing to install Maximo Application Suite on Amazon Web Services.

This product requires an internet connection to deploy properly. The following code is downloaded on deployment for setting up the Maximo Application Suite Red Hat OpenShift Container Platform cluster:

- https://github.com/ibm-mas/ansible-devops.git
- https://github.com/ibm-mas/multicloud-bootstrap.git

About this task

The Maximo Application Suite client-managed solution is available with two different Marketplace products.

Maximo Application Suite client-managed with Red Hat OpenShift entitlement
This product includes the subscription for the Red Hat OpenShift cluster that is deployed during the installation process.

The following CloudFormation templates are available as fulfillment options:

- A new Red Hat OpenShift cluster by using the Installer Provisioned Infrastructure (IPI)
- A new Red Hat OpenShift cluster by using the User Provisioned Infrastructure (UPI)

Maximo Application Suite client-managed without Red Hat OpenShift entitlement
This product does not include the subscription for the Red Hat OpenShift cluster that is deployed during the installation process or the existing one provided by the user. You must have your own Red Hat OpenShift subscription to deploy Maximo Application Suite by using this product.

The following CloudFormation templates are available as fulfillment options:

- An existing Red Hat OpenShift cluster
- A new Red Hat OpenShift cluster by using the Installer Provisioned Infrastructure (IPI)
- A new Red Hat OpenShift cluster by using the User Provisioned Infrastructure (UPI)

After you select the Paid product from Amazon Web Services Marketplace, you can complete the following steps for a public paid offering. The public paid offer is the default option that you see after you view the product page.

Procedure

1. In the Amazon Web Services Marketplace service console, depending on if you have the Red Hat OpenShift subscription or not, click **Discover product** and search for Maximo Application Suite client-managed without Red Hat OpenShift entitlement or Maximo Application Suite client-managed with Red Hat OpenShift entitlement. The products are sold by IBM Maximo.
2. Open Maximo Application Suite.
3. Review the product information and click **Continue to Subscribe.**

4. On the **Create an agreement for this software** page, select any one of the contract duration options: 12 months / 24 months / 36 months.
5. Select a minimum of **500 AppPoints** and click **Create Contract**.
6. After 24 hours of subscription, to register the product with your Amazon Web Services account, complete the registration by using an IBMid from https://www.ibm.com/marketplace/connector/landing/aws/services/amicontractsetup.

   You receive an email notification in few minutes to obtain the artifacts required for the product deployment.
7. After you retrieve the entitlement key, link your Maximo Application Suite subscription with Red Hat OpenShift account by following the steps at https://www.ibm.com/docs/en/cloud-paks/1.0?topic=iocpc-accessing-red-hat-entitlements-from-your-cloud-paks, obtain the Red Hat OpenShift pull secret from the Red Hat OpenShift account, return to product page, and click **Continue to Configuration**.

   Note: Linking the Maximo Application Suite subscription with Red Hat OpenShift account is required only if you are deploying Maximo Application Suite with new Red Hat OpenShift cluster. if you decide to use existing Red Hat OpenShift cluster, it is not required.
8. In **Fulfillment** option, select the appropriate deployment mode. For Maximo Application Suite deployment with new Red Hat OpenShift cluster, you can select either new network infrastructure (IPI), existing network infrastructure (UPI), or existing Red Hat OpenShift cluster. For more information, refer to the fulfillment options explained earlier in this section.
9. In **Software version**, select the latest version.
10. In **Region**, select a supported geographical region where you want to install the Maximo Application Suite. For the list of supported regions, see Preparing to install Maximo Application Suite on Amazon Web Services.
11. Accept the default values in the other fields and click **Continue to Launch**.
12. In the **Launch** page, select **Choose Action** > **Launch CloudFormation**.
13. To open the CloudFormation stack wizard, click **Launch**.
14. In the **Create stack** wizard step, accept the default values, and click **Next**.
15. In the **Specify stack details** step, in the **Stack name** section, enter a unique name.
16. In the Parameters section, enter the installation parameters by using the information that you gathered when you configured the Prerequisites for installing Maximo Application Suite on Amazon Web Services and Preparing to install Maximo Application Suite on Amazon Web Services.
17. Enter all of the mandatory parameters.
    - The Maximo Application Suite offering type, such as Maximo Application Suite Core and Cloud Pak for Data or Maximo Application Suite and Maximo Manage.

      Note: If you choose to install Maximo Application Suite with Manage, you can use the default IBM Db2 instance that is provisioned by IBM instead of configuring your own external Db2 instance. To configure the default IBM Db2, do not add information in the username, password, JDBC URL, certificate URL, and demo data fields.

      The internal Db2 configuration is available from Maximo Application Suite 8.10 or later.

      Starting in 8.11, for US GovCloud regions, Db2 is not supported.
    - All of the parameters in the **Cluster and bootnode access** and **Keys and licenses** sections.
18. To create a new cluster, complete the New OpenShift cluster configuration details section.
19. In the optional parameter groups, such as the group of Maximo Manage database configuration parameters, ensure that you either specify all of the parameter values or leave all of them empty.
20. To configure Amazon Managed Streaming for Kafka, select **Yes**.

    The Amazon Managed Streaming for Kafka is configured to process data streaming of applications such as IoT and IBM Maximo Monitor from Maximo Application Suite.

    You can configure the default Amazon Managed Streaming from Maximo Application Suite 8.10. or later.

21. To configure a DocumentDB instance, you can select any of the following options.
    - Configure a new MongoDB community edition
    - Use an existing MongoDB

      You must input existing MongoDB connection details such as the username, password, MongoDB hostname, and CA certificate.
    - Configure a new Amazon DocumentDB
    - Use an existing Amazon DocumentDB

      Add existing DocumentDB connection details such as the username, password, DocumentDB hostname, and CA certificate.

      You must input the VPC ID of the region in which the Amazon DocumentDB is deployed. Ensure that the DocumentDB does not have a matching or overlapping `IPv4 CIDR block 10.0.0.0/16`.

    You can configure the default DocumentDB from Maximo Application Suite 8.10. or later.
22. In the **Other parameters**, select the **OperationalMode** as `Production` or `Non-production`.

    You can specify deployments for Production or Non-production environments. Non-production installations can be used for internal development and testing. The installation AppPoints are unused in the Non-production installations. These specifications are also visible in the metrics shared with IBM as well as on the product UI.
23. Click **Next**.
24. In the **Configure stack options** step, configure any additional options that you require. To know more about the stack options, click the "Learn more" links available in each option. Click **Next** when done.
25. In the **Review** step, review the values that you entered and acknowledge the message that relates to identity and access management (IAM) resources.
26. To begin the installation, click **Create stack**.

What to do next

During the installation process, the AWS CloudFormation stack template that you configured is used to create a Bootnode. The Bootnode contains all required resources to complete the installation. To verify that the Bootnode is created successfully, in the **CloudFormation** > **Stacks** page, confirm that the stack status is updated to `CREATE_IN_PROGRESS`

For more information, see “Monitoring IBM Maximo Application Suite installation on Amazon Web Services” on page 48.

Installing client managed IBM Maximo Application Suite for private paid offer

The paid AWS product can be installed from the AWS Marketplace based on your infrastructure needs. A new Red Hat OpenShift instance is created as part of the Maximo Application Suite installation.

Before you begin

Before you can install Maximo Application Suite on Amazon Web Services, you must configure prerequisites and gather information that you need to complete the installation.

For more information, see Prerequisites for installing Maximo Application Suite on Amazon Web Services.

Consider other criteria, such as the type of Maximo Application Suite offering that you want and whether you want to create a Red Hat OpenShift cluster or reuse an existing one.

Note: The existing cluster must be created by using the automated deployment option only.

For more information, see Preparing to install Maximo Application Suite on Amazon Web Services.

This product requires an internet connection to deploy properly. The following code is downloaded on deployment for setting up the Maximo Application Suite Red Hat OpenShift Container Platform cluster:

- https://github.com/ibm-mas/ansible-devops.git

- https://github.com/ibm-mas/multicloud-bootstrap.git

About this task

The Maximo Application Suite client-managed solution is available with two different Marketplace products.

Maximo Application Suite client-managed with Red Hat OpenShift entitlement
This product includes the subscription for the Red Hat OpenShift cluster that is deployed during the installation process.

The following CloudFormation templates are available as fulfillment options:

- A new Red Hat OpenShift cluster by using the Installer Provisioned Infrastructure (IPI)
- A new Red Hat OpenShift cluster by using the User Provisioned Infrastructure (UPI)

Maximo Application Suite client-managed without Red Hat OpenShift entitlement
This product does not include the subscription for the Red Hat OpenShift cluster that is deployed during the installation process or the existing one provided by the user. You must have your own Red Hat OpenShift subscription to deploy Maximo Application Suite by using this product.

The following CloudFormation templates are available as fulfillment options:

- An existing Red Hat OpenShift cluster
- A new Red Hat OpenShift cluster by using the Installer Provisioned Infrastructure (IPI)
- A new Red Hat OpenShift cluster by using the User Provisioned Infrastructure (UPI)

After you select the Paid product from Amazon Web Services Marketplace, you can complete the following steps for a private paid offering.

You can avail a private offer for IBM Maximo Application Suite subscription contract with custom configuration and pricing by contacting a IBM sales representative or viewing https://www.ibm.com/products/maximo/pricing page. After discussions and agreements, the IBM sales representatives share the offer URL page link.

Procedure

1. Click the private URL shared by the IBM sales representatives, and click **Continue to Subscribe**.
2. Review the number of AppPoints, contract duration, and contract pricing information for the private offer, and click **Create Contract**.
3. Confirm the contract terms on the confirmation dialog.

   You see a message that AWS is processing the request. It takes couple of minutes to complete the processing. Until then, **Continue to Configuration** is unavailable and for **Create Contract** displays the label as Pending.
4. After you confirm the contract terms of purchase to register the product with your AWS account, complete the registration by using an IBM ID from https://www.ibm.com/marketplace/connector/landing/aws/services/amicontractsetup. You can also get this URL from the Overview section of the product page.

   Select the Maximo Application Suite product for which you subscribed, provide the necessary details like the AWS account number, and confirm the pending order by verifying the quotation number. You see the product summary page for your purchase. The browser tab can be closed. You receive an email notification within few minutes to obtain the IBM entitlement key from My IBM portal required for the product deployment.
5. After you retrieve the entitlement key, link your IBM Maximo Application Suite subscription with Red Hat OpenShift account by following the steps at https://www.ibm.com/docs/en/cloud-paks/1.0?topic=iocpc-accessing-red-hat-entitlements-from-your-cloud-paks, obtain the Red Hat OpenShift pull secret from the Red Hat OpenShift account, return to product page, and click **Continue to Configuration**.

Note: Linking the IBM Maximo Application Suite subscription with Red Hat OpenShift account is required only if you are deploying IBM Maximo Application Suite with new Red Hat OpenShift cluster. It is not required if you decide to use existing Red Hat OpenShift cluster. (Support for existing Red Hat OpenShift cluster is not available yet.)

6. In **Fulfillment** option, select the appropriate deployment mode.

   For Maximo Application Suite deployment with new Red Hat OpenShift cluster, you can select either new network infrastructure (IPI) or existing network infrastructure (UPI), or existing Red Hat OpenShift cluster. For more information, refer to the fulfillment options explained earlier in this section.
7. In **Software version**, select the latest version.
8. In **Region**, select a supported geographical region where you want to install the Maximo Application Suite.

   For the list of supported regions, see Preparing to install Maximo Application Suite on Amazon Web Services.
9. Accept the default values in the other fields and click **Continue to Launch**.
10. In the **Launch** page, select **Choose Action** > **Launch CloudFormation**.
11. To open the CloudFormation stack wizard, click **Launch**.
12. In the **Create stack** wizard step, accept the default values, and click **Next**.
13. In the **Specify stack details** step, in the **Stack name** section, enter a unique name.
14. In the Parameters section, enter the installation parameters by using the information that you gathered when you configured the Prerequisites for installing Maximo Application Suite on Amazon Web Services and Preparing to install Maximo Application Suite on Amazon Web Services.
15. Enter all of the mandatory parameters.
    - The Maximo Application Suite offering type, such as Maximo Application Suite Core and Cloud Pak for Data or Maximo Application Suite and Maximo Manage.

      Note: If you choose to install Maximo Application Suite with Manage, you can use the default IBM Db2 instance that is provisioned by IBM instead of configuring your own external Db2 instance. To configure the default IBM Db2, do not add information in the username, password, JDBC URL, certificate URL, and demo data fields.

      The internal Db2 configuration is available from Maximo Application Suite 8.10 or later.
    - All of the parameters in the **Cluster and bootnode access** and **Keys and licenses** sections.
16. To create a new cluster, complete the New OpenShift cluster configuration details section.
17. In the optional parameter groups, such as the group of Maximo Manage database configuration parameters, ensure that you either specify all of the parameter values or leave all of them empty.
18. To configure Amazon Managed Streaming for Kafka, select **Yes**.

    The Amazon Managed Streaming for Kafka is configured to process data streaming of applications such as IoT and IBM Maximo Monitor from Maximo Application Suite.

    You can configure the default Amazon Managed Streaming from Maximo Application Suite 8.10. or later.
19. To configure a DocumentDB instance, you can select any of the following options.
    - Configure a new MongoDB community edition
    - Use an existing MongoDB

      You must input existing MongoDB connection details such as the username, password, MongoDB hostname, and CA certificate.
    - Configure a new Amazon DocumentDB
    - Use an existing Amazon DocumentDB

      Add existing DocumentDB connection details such as the username, password, DocumentDB hostname, and CA certificate.

You must input the VPC ID of the region in which the Amazon DocumentDB is deployed. Ensure that the DocumentDB does not have a matching or overlapping `IPv4 CIDR block 10.0.0.0/16`.

You can configure the default DocumentDB from Maximo Application Suite 8.10. or later.

20. In the **Other parameters**, select the **OperationalMode** as `Production` or `Non-production`.

    You can specify deployments for Production or Nonproduction environments. Nonproduction installations can be used for internal development and testing. The installation AppPoints are unused in the Non-production installations. These specifications are also visible in the metrics shared with IBM as well as on the product UI.
21. Click **Next**.
22. In the **Configure stack options** step, configure any additional options that you require . To know more about the stack options, click the "Learn more" links available in each option. Click **Next** when done.
23. In the **Review** step, review the values that you entered and acknowledge the message that relates to identity and access management (IAM) resources.
24. To begin the installation, click **Create stack**.

What to do next

During the installation process, the AWS CloudFormation stack template that you configured is used to create a Bootnode. The Bootnode contains all required resources to complete the installation. To verify that the Bootnode is created successfully, in the **CloudFormation** > **Stacks** page, confirm that the stack status is updated to `CREATE_IN_PROGRESS`

For more information, see “Monitoring IBM Maximo Application Suite installation on Amazon Web Services” on page 48.

Installing Maximo Application Suite with Red Hat OpenShift on Amazon Web Services

Starting in IBM Maximo Application Suite 8.11, install the application with Red Hat OpenShift on Amazon Web Services (AWS). You can deploy Maximo Application Suite from the AWS marketplace by using the BYOL or paid offerings so that you can use your existing Red Hat OpenShift cluster.

Before you begin

Create the Virtual Private Cloud (VPC) and subnets for ROSA cluster by using the `pre-req-vpc-subnets.sh` file. Download the compressed file from IBMGitHub UPI resources page. Refer to the `readme.txt` file in the compressed file folder for instructions to run the `pre-req-vpc-subnets.sh` file.

Procedure

1. Install the ROSA cluster by following the steps in Getting started.

   The steps include running the command to create a ROSA cluster, giving the cluster a name, selection the option **Install** on the existing VPC and subnet, and selecting VPC and subnet that was created.

   Tip: Make sure that you create a public cluster with Federal Information Processing Standards (FIPS) that is not enabled.
2. Create an Amazon Elastic Files System (EFS) storage by running the **`ocp_efs`** ansible role.

   Before you run this ansible role, append the ROSA cluster name to the VPC name used in ROSA cluster creation.

   For example, if the VPC name for ROSA cluster creation is *mas-vpc* and the ROSA cluster name is *samplerosacluster*, then the VPC name is *mas-vpc-samplerosacluster*.
3. Run the following commands to set required environment parameters that are necessary to run `ocp_efs` ansible role.

```
export AWS_ACCESS_KEY_ID=<your aws access key id>
export AWS_SECRET_ACCESS_KEY=<your aws access key>
```

```
export AWS_DEFAULT_REGION=<aws region where ROSA cluster is created>
export CLUSTER_NAME=<ROSA cluster name>
```

4. Connect to your ROSA cluster by running the **oc login** command.
   The following is an example of the **oc login** command:

   ```
   oc login -\-token=<your_server_token> -\-server=https://<server_host>:<port_number>
   ```

   Tip: You can obtain this login command information in your ROSA cluster web console by clicking **cluster-admin** > **Copy login command**.
5. Run the following commands to install the ansible collection and **ocp_efs** ansible role.

   ```
   ansible-galaxy collection install ibm.mas_devops
   export ROLE_NAME=ocp_efs && ansible-playbook ibm.mas_devops.run_role
   ```

   Make sure that the EFS storage is created on the Amazon Web Services console. Verify that the storage class is created on the ROSA cluster console. After the verification, proceed with the stack deployment by completing the **CloudFormation** template page fields.
6. Follow the **Existing Red Hat OpenShift cluster** option available under the IBM Maximo Application Suite (BYOL) AWS marketplace listing to deploy Maximo Application Suite core and IBM Cloud Pak for Data stack or Maximo Application Suite core and IBM Maximo Manage stack.

   Note: Only **Existing Red Hat OpenShift cluster** option is supported for ROSA.
7. Enter the required input parameters on **CloudFormation** page to create the stack.
   For more information, see "Installing BYOL IBM Maximo Application Suite" on page 33.

Installing Cloud Pak for Data on an Amazon Web Services instance of Maximo Application Suite

To install Cloud Pak for Data on an existing Red Hat OpenShift cluster instance that has been configured to run the automation on Amazon Web Services, you clone a GitHub repository and run a script.

For example, install Cloud Pak for Data on an Amazon Web Services instance on a cluster that has been installed with Maximo Application Suite and Maximo Manage offerings, not on an existing cluster with Cloud Pak for Data already installed.

The following components will be installed:

- Cloud Pak for Data
- Db2 Warehouse services
- Data Management Console

If you installed Maximo Application Suite and Maximo Manage offerings, your Red Hat OpenShift cluster does not include Cloud Pak for Data. If you want to deploy certain applications and add-ons in the Maximo Application Suite, such as Maximo Predict, you must first install Cloud Pak for Data.

To install Cloud Pak for Data, you clone a Git repository, locate the installation script, and run it. When you run the script, you must provide the AWS CloudFormation stack values that identify the Red Hat OpenShift cluster, such as its region, its stack name, and your entitled registry key. The installation script retrieves the Maximo Application Suite <unique-string> and the Red Hat OpenShift cluster details from the stack. If the cluster credentials, that is the username and password, are changed since you installed the Maximo Application Suite, you must provide the updated credentials when you run the script.

You run the script on your local machine or on the Bootnode in the Maximo Application Suite Red Hat OpenShift cluster. For more information, see "Accessing the Bootnode and Red Hat OpenShift cluster" on page 55.

Note: Starting in 8.11, for US GovCloud regions, the Cloud Pak for Data configuration is not supported.

Before you begin

1. Retrieve the entitled registry key that you provided when you installed the Maximo Application Suite.

If you do not have this key, use the steps in the MAS download document to download this key from the IBM Container Library.

2. In the AWS CloudFormation console, in the **CloudFormation->Stacks** page, locate the stack that you created when you installed your Red Hat OpenShift cluster.
3. From the **Outputs** tab, record the following values:
   - The instance's stack name
   - OpenShiftConsoleUrl (ocp url)
   - OpenShiftPassword (ocp password)
   - OpenShiftUser (ocp username)
4. In the menu bar, record the region value.
5. On the machine where you want to run the script, ensure that the following CLI packages are installed:
   - Version 4.0 or a later version of GNU bash
   - jq
   - Git
   - AWS CLI. Ensure that this package is configured for authentication with your AWS account. For more information, see Configuring the AWS CLI in the AWS documentation.

Procedure

Complete the following steps to install Cloud Pak for Data in your Red Hat OpenShift cluster:

1. On the machine where you want to run the script, in a command shell, log in to the AWS service by running the following command:

   ```
   aws configure
   ```

   You are prompted for your identity and access management (IAM) user credentials. Enter the credentials for an IAM user that has the permissions to run the script, such as the IAM user that installed the Maximo Application Suite. For more information, see Configuring the installation permissions.
2. Clone the Git repository that contains the script by running the following command:

   ```
   git clone https://github.com/ibm-mas/mas-on-aws.git
   ```
3. Make the script executable by running the following commands:

   ```
   cd mas-on-aws
   chmod +x deploy-cp4d.sh
   ```
4. View the script's usage information by running the following command:

   ```
   ./deploy-cp4d.sh -h
   ```
5. Specify the required options and run the script.
   - Use the `r` option to specify the region code of the region where the Maximo Application Suite was installed, for example: `-r ap-northeast-3`
   - Use the `s` option to specify the Maximo Application Suite CloudFormation stack name, for example: `-s sp-manage-12`
   - Use the e option to specify the entitled registry key that you provided when you installed the Maximo Application Suite, for example: `-e <entitlement-key>`
   - Use the u and p options to specify the Maximo Application Suite Red Hat OpenShift cluster credentials, for example: `-u <ocp-user> -p <ocp-password>`

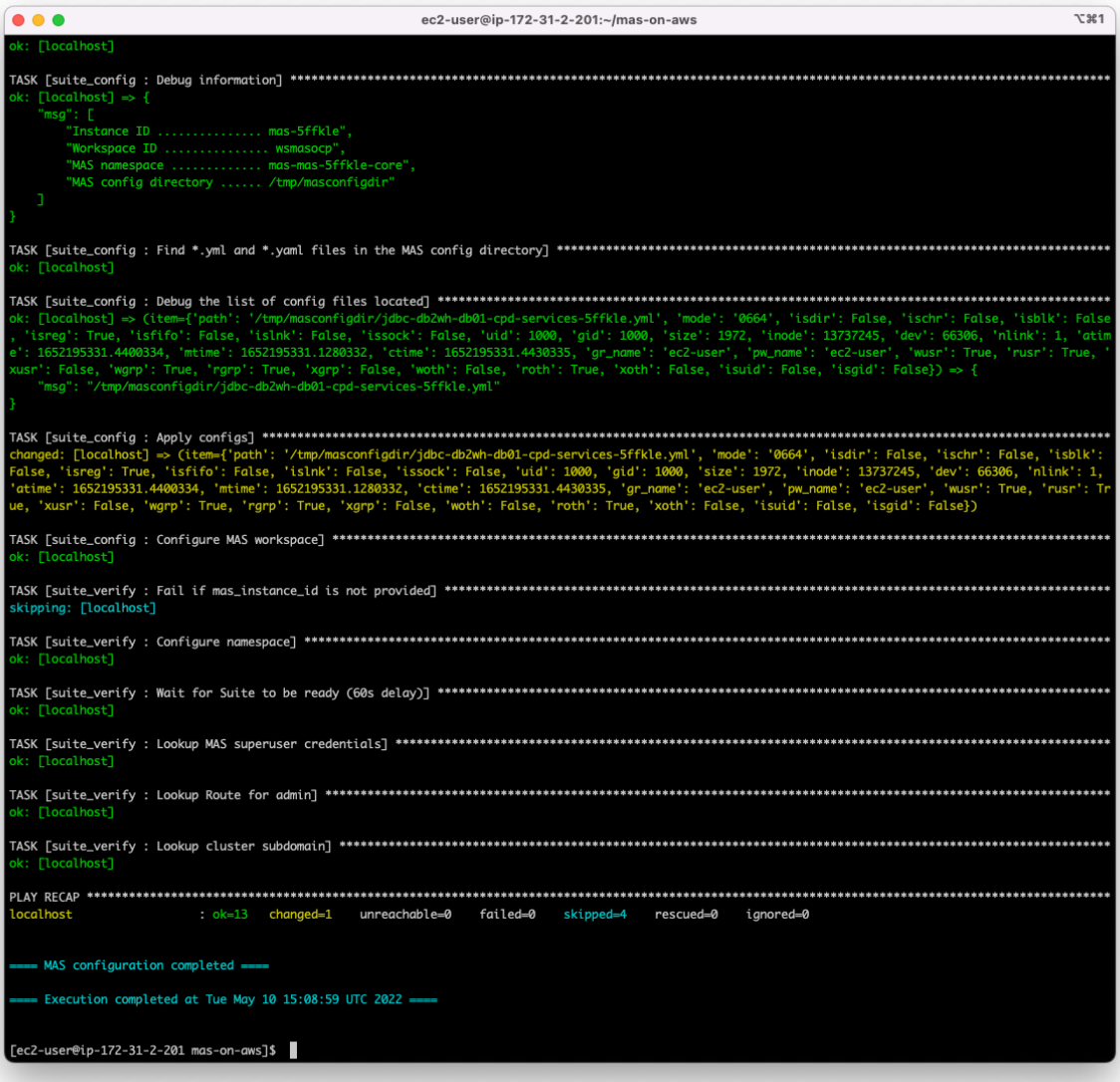
- The following sample command installs Cloud Pak for Data by using all of these example options:

```
./deploy-cp4d.sh -r ap-northeast-3 -s sp-manage-12 -e <entitlement-key> -u <ocp-user> -p
<ocp-password>
```

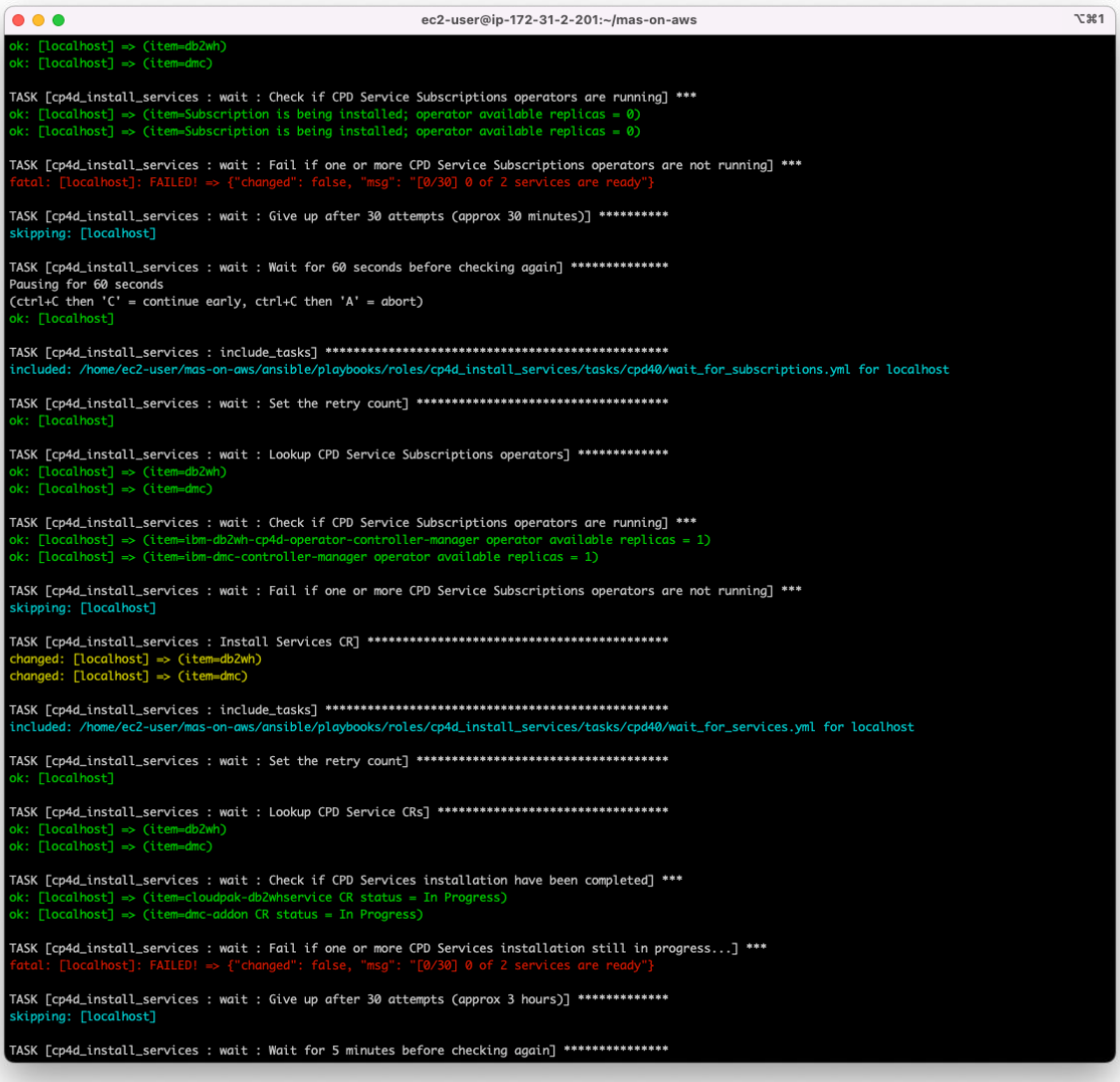
The script takes 60 minutes to install Cloud Pak for Data into the Red Hat OpenShift cluster.

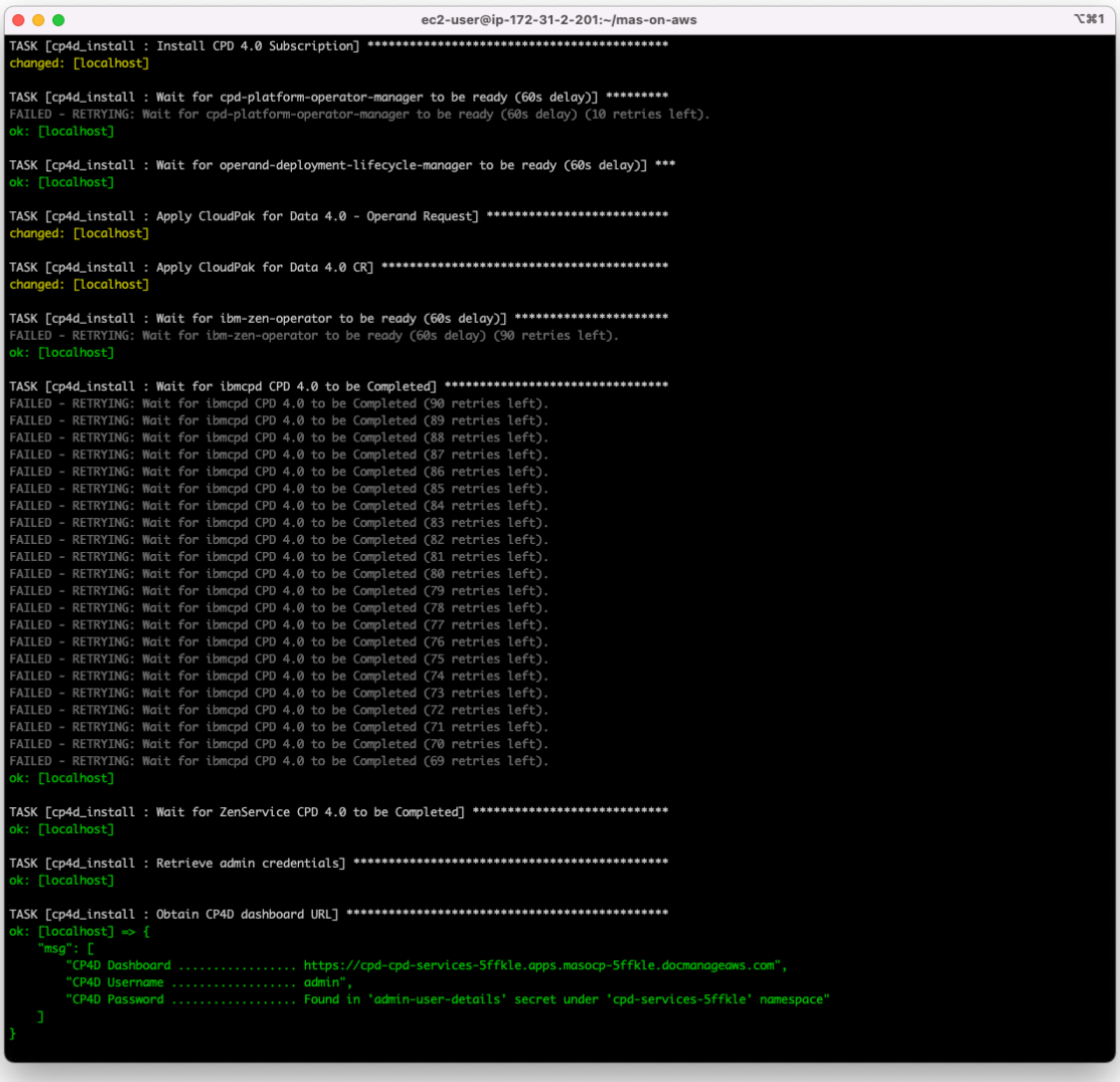
For reference, the following screen shots show what the script output looks like during this process:

When the script is running:

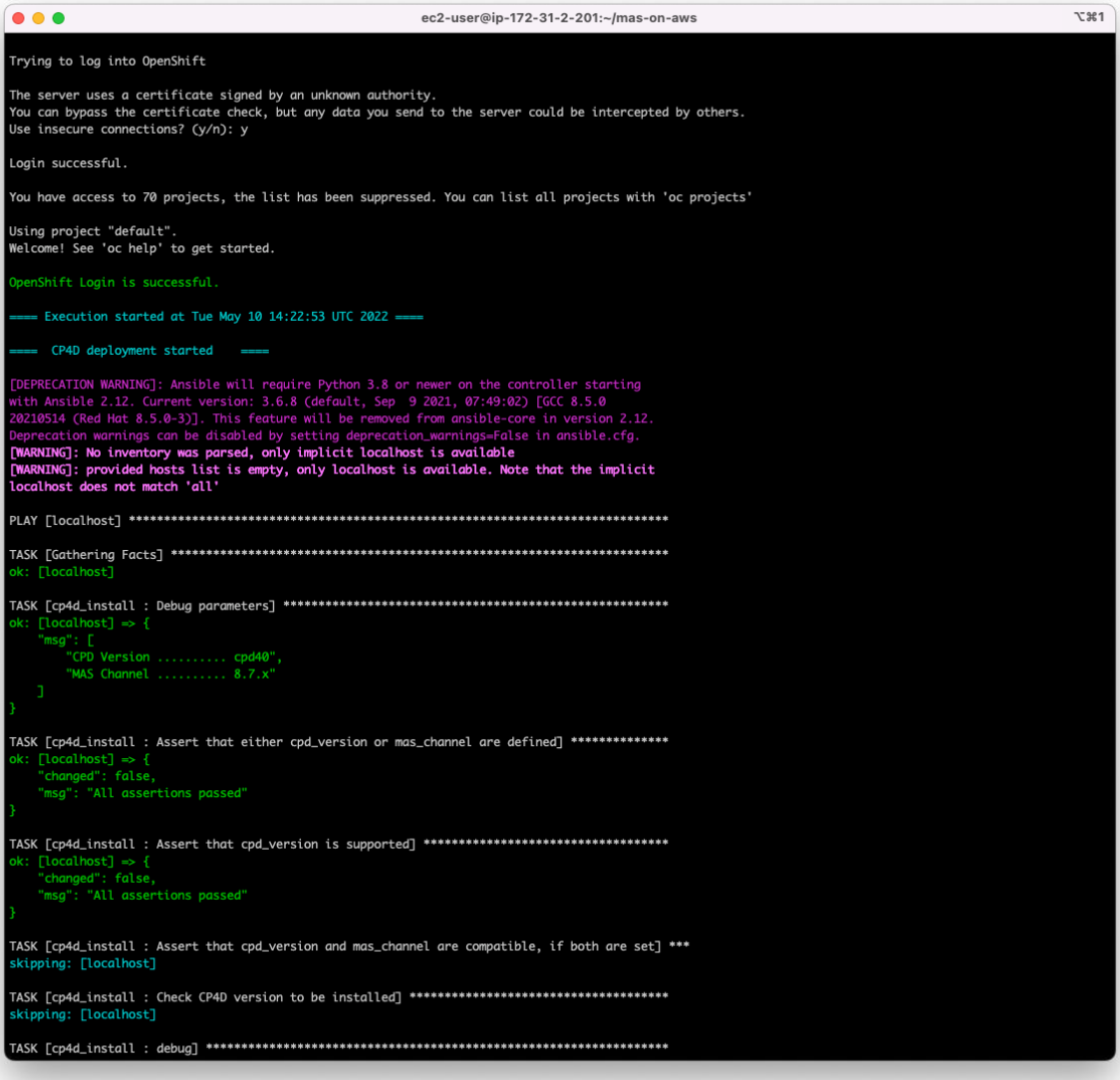


You may see some **FAILED** messages and **fatal** messages. These messages are normal and are automatically solved during the script execution:


ec2-user@ip-172-31-2-201:~/mas-on-aws
⌥⌘1
ok: [localhost] => (item=db2wh)
ok: [localhost] => (item=dmc)
TASK [cp4d_install_services : wait : Check if CPD Service Subscriptions operators are running] ***
ok: [localhost] => (item=Subscription is being installed; operator available replicas = 0)
ok: [localhost] => (item=Subscription is being installed; operator available replicas = 0)
TASK [cp4d_install_services : wait : Fail if one or more CPD Service Subscriptions operators are not running] ***
fatal: [localhost]: FAILED! => {"changed": false, "msg": "[0/30] 0 of 2 services are ready"}
TASK [cp4d_install_services : wait : Give up after 30 attempts (approx 30 minutes)] **********
skipping: [localhost]
TASK [cp4d_install_services : wait : Wait for 60 seconds before checking again] *************
Pausing for 60 seconds
(ctrl+C then 'C' = continue early, ctrl+C then 'A' = abort)
ok: [localhost]
TASK [cp4d_install_services : include_tasks] **
included: /home/ec2-user/mas-on-aws/ansible/playbooks/roles/cp4d_install_services/tasks/cpd40/wait_for_subscriptions.yml for localhost
TASK [cp4d_install_services : wait : Set the retry count] **********************************
ok: [localhost]
TASK [cp4d_install_services : wait : Lookup CPD Service Subscriptions operators] ************
ok: [localhost] => (item=db2wh)
ok: [localhost] => (item=dmc)
TASK [cp4d_install_services : wait : Check if CPD Service Subscriptions operators are running] ***
ok: [localhost] => (item=ibm-db2wh-cp4d-operator-controller-manager operator available replicas = 1)
ok: [localhost] => (item=ibm-dmc-controller-manager operator available replicas = 1)
TASK [cp4d_install_services : wait : Fail if one or more CPD Service Subscriptions operators are not running] ***
skipping: [localhost]
TASK [cp4d_install_services : Install Services CR] ***
changed: [localhost] => (item=db2wh)
changed: [localhost] => (item=dmc)
TASK [cp4d_install_services : include_tasks] **
included: /home/ec2-user/mas-on-aws/ansible/playbooks/roles/cp4d_install_services/tasks/cpd40/wait_for_services.yml for localhost
TASK [cp4d_install_services : wait : Set the retry count] **********************************
ok: [localhost]
TASK [cp4d_install_services : wait : Lookup CPD Service CRs] *******************************
ok: [localhost] => (item=db2wh)
ok: [localhost] => (item=dmc)
TASK [cp4d_install_services : wait : Check if CPD Services installation have been completed] ***
ok: [localhost] => (item=cloudpak-db2whservice CR status = In Progress)
ok: [localhost] => (item=dmc-addon CR status = In Progress)
TASK [cp4d_install_services : wait : Fail if one or more CPD Services installation still in progress...] ***
fatal: [localhost]: FAILED! => {"changed": false, "msg": "[0/30] 0 of 2 services are ready"}
TASK [cp4d_install_services : wait : Give up after 30 attempts (approx 3 hours)] ************
skipping: [localhost]
TASK [cp4d_install_services : wait : Wait for 5 minutes before checking again] **************



Finally, when the script is completed successfully:



6. After the installation, obtain the Cloud Pak for Data instance URL and credentials from the OpenShift console.

   One of the many ways to obtain these are:

   - Go to **Projects**, and click the Cloud Pak for Data project name which follows the format `cpd-services-<unique-string>`.
   - In the **Overview** tab, under **Inventory**, click **Routes.**
   - The URL for the Cloud Pak for Data instance is under the "Location" column for the route named `cpd`.
   - In the same **Overview** and **Inventory** section click **Secrets**.
   - Search for `admin-user-details`, click it, and scroll down the page to see the value for `initial_admin_password`. This is the password to be able to login the Cloud Pak for Data URL. Use `admin` as the username.

What to do next

You can now install and configure dependencies inside Cloud Pak for Data that are required for some Maximo Application Suite applications. For more information about the complete list of prerequisites for each application, see Prerequisite software.

You also can enable the internet access to a Db2 Warehouse database instance in the Amazon Web Services Red Hat OpenShift cluster. For more information, see Enabling internet access to a Db2 Warehouse database on an AWS Red Hat OpenShift cluster.

Monitoring IBM Maximo Application Suite installation on Amazon Web Services

During the installation process, the AWS CloudFormation stack template that you configured is used to create a Bootnode. The Bootnode contains all required resources to complete the installation. To verify that the Bootnode is created successfully, in the **CloudFormation** > **Stacks** page, confirm that the stack status is updated to `CREATE_IN_PROGRESS`.

Before you begin

To ensure that the Red Hat OpenShift cluster is created, the Bootnode starts a bootstrap process. This process creates a bootstrap node that uses the Red Hat OpenShift installer to create master and worker nodes. To verify that the bootstrap process is started, in the **CloudFormation** > **Stacks** page, click the **Events** tab. When the `DeployWaitCondition` event is displayed, the bootstrap process is started.

About this task

You can monitor the progress of the remaining installation tasks in the installation logs. These logs can be viewed in the AWS CloudWatch service or the Bootnode.

Procedure

1. Monitoring the installation logs in AWS CloudWatch.

   To monitor the installation progress in AWS CloudWatch, complete the following steps:

   a) In the AWS CloudWatch management console, click **Logs**.

   b) In the log group list, click the group that is named `/ibm/mas/masocp-<unique-string>`.

   c) In the log stream list, open the installation log by clicking the stream that is named `mas-provisioning-logs`.

   d) Optional: In the installation log stream, if the message `Auto retry paused` is displayed, click the `Resume` link to display the latest log updates.

2. Monitoring the installation logs in the Bootnode.

   To monitor the installation progress in the Bootnode, complete the following steps:

   a) Connect to the Bootnode by using Secure Shell (SSH) access. For instructions, see Accessing the Bootnode and Red Hat OpenShift cluster.

   b) Run the following command to switch to the root user:

   ```
   sudo su -
   ```

   c) Monitor the installation log updates by running the following command:

   ```
   tail -f /root/ansible-devops/multicloud-bootstrap/mas-provisioning.log
   ```

3. After the installation is completed, in the **CloudFormation** > **Stacks** page, the following indicators confirm that the installation is successful:

   - The status of the stack is **CREATE_COMPLETE**.
   - In the **Outputs** page, the `DeploymentStatus` parameter displays a message that indicates that the installation succeeded, for example `ID-aws-small-masocp-fjh2sx:SUCCESS#MAS deployment completed successfully`.

4. Proactively check the status of the deployment.

   Depending on the parameters that you specified, the installation time might vary.

If the installation is unsuccessful, use the information in the Troubleshooting installation problems topic to identify and resolve the problem.

Accessing IBM Maximo Application Suite

After you install IBM Maximo Application Suite, to access it, you need the administrator URL, user credentials, and public certificate.

About this task

You must have the following information to access the Maximo Application Suite:

- The Maximo Application Suite administrator URL, which you use to connect to the Maximo Application Suite through a browser.
- Your username and password.
- The public certificate for the Maximo Application Suite.

  You import this certificate into your browser's trusted store to ensure secure communication between your browser and Maximo Application Suite.

How you retrieve these items of information depends on whether you configured a verified Amazon SES email address.

If you have a verified Amazon SES email address, you can retrieve the administrator URL, username, and password from the emails that you received. The public certificate is attached to these emails.

If you do not have an Amazon SES email address, you can retrieve the administrator URL from the CloudFormation console for the stack that you created during the installation. However, to retrieve your username, password, and the public certificate, you must connect to the Red Hat OpenShift cluster.

Product versions

Following products are installed as part of the Maximo Application Suite installation. If you provide the existing Red Hat OpenShift cluster, the installation process checks if any of these products are already installed. For details about the installation process behavior if existing products are found, see Preparing to install Maximo Application Suite on Amazon Web Services.

- Red Hat OpenShift 4.15.x

  Supported on Installer Provisioned Infrastructure (IPI), User Provisioned Infrastructure (UPI), and existing Red Hat OpenShift cluster deployment.
- Red Hat OpenShift Service on AWS (ROSA) 4.15.x

  Supported on existing Red Hat OpenShift cluster deployment on Amazon Web Services public cloud.
- IBM Cloud Pak foundational services 4.7.0
- IBM Cloud Certificate Manager 3.25.13
- MongoDB (CE) 7.0.12
- IBM Suite License Service 3.10.1
- IBM Data Reporter Operator 2.18.0
- IBM Cloud Pak for Data 6.0.0
- IBM Maximo Application Suite 9.0.5
- IBM Maximo Manage 9.0.5

Procedure

1. In the **Outputs** section of the CloudFormation stack that was created during the deployment, record the following values:
   - The value of the `masAdminUrl` key. This value contains the administrator URL.

- The values of the `openShiftConsoleUrl`. These values contain the URL for the Red Hat OpenShift console.

  Note: The `openShiftConsoleUrl` is not displayed for an existing Red Hat OpenShift Container Platform.
- The value of the `clusterUniqueString` key. This value contains the cluster unique string to look for the correct resources in Red Hat OpenShift.

2. Retrieve the Red Hat OpenShift Container Platform credentials from the `maximo-ocp-secret`.

   The Amazon Web Services Secrets Manager secret contains the credentials for Red Hat OpenShift cluster. It consists of a secret named `maximo-ocp-secret` containing Red Hat OpenShift credentials.
3. Retrieve the Red Hat OpenShift Container Platform `kubeadmin` credentials from `maximo-kubeadmin-secret`.
4. Connect to the Red Hat OpenShift cluster.
5. Select **Workloads** > **Secrets** from the navigation page.
6. Select the project named `mas-<unique-string>-core`.
7. Click the secret that is named `<unique-string>-credentials-superuser`.
8. Click the 'Reveal values' link to get the username and password for Maximo Application Suite.
9. Click the secret that is named `<unique-string>-cert-public` from the `mas-<unique-string>-core` project.
10. Click the 'Reveal values' link to get the contents of the certificates.
11. Retrieve the contents of `ca.crt`, which is the public certificate for Maximo Application Suite.
12. After you import the public certificate into your browser's trusted store, paste the Maximo Application Suite administrator URL into your browser, and enter the authentication credentials to access the application.
13. Log in to the Maximo Application Suite to deploy applications, create users, and specify configuration.

    To resolve loading issues with the initial setup page, you can provision self-signed certificates that are needed to add in the used browser's truststore. For more information, see IBM Maximo Application Suite - Initial setup page does not load.

What to do next

If you installed the Maximo Application Suite core and Maximo Manage, and you want to deploy certain applications and add-ons in the Maximo Application Suite, which requires Cloud Pak for Data, see “Installing Cloud Pak for Data on an Amazon Web Services instance of Maximo Application Suite” on page 42.

If you installed Maximo Application Suite core and Cloud Pak for Data, Cloud Pak for Data is already installed for you, however you might still need to install some Cloud Pak for Data services based on the applications you want to deploy.

You also can enable internet access to a Db2 Warehouse database instance in the AWS Red Hat OpenShift cluster. For more information, see Enabling internet access to a Db2 Warehouse database on an AWS Red Hat OpenShift cluster.

If you want to use well-known certificates that are signed by certificate authority as **Let's Encrypt**, it is recommended that you uninstall this default Maximo Application Suite instance and install another after you follow the next steps to configure Let's Encrypt and Route53 on AWS.

Note: You cannot install Maximo Application Suite core and Cloud Pak for Data over a cluster that is deployed through Maximo Application Suite core or Maximo Manage.

For more information, see Prerequisite software and Deploying applications, add-ons and industry solutions.

Configuring Let's Encrypt for Maximo Application Suite on Amazon Web Services

When you install Maximo Application Suite along with a Stack on Amazon Web Services using the automated deployment offerings, Maximo Application Suite uses self-signed certificates.

If you want to use well-known certificates that are signed by Certificate Authority such as Let's Encrypt, install and configure Let's Encrypt and Route53 on Amazon Web Services.

Before you begin

Complete the following tasks:

1. Create an Access Key in the Amazon Web Services console.
2. Create an IAM policy so that certificate manager is able to add records to Route53 in order to solve the DNS01 challenges. To create an IAM policy, complete the following steps:
   a. Login to the Amazon Web Services console, then search for IAM and click the first option that is displayed.
   b. Click **Policies** and then click **Create Policy**.
   c. Go to the **JSON** tab and paste the following JSON:

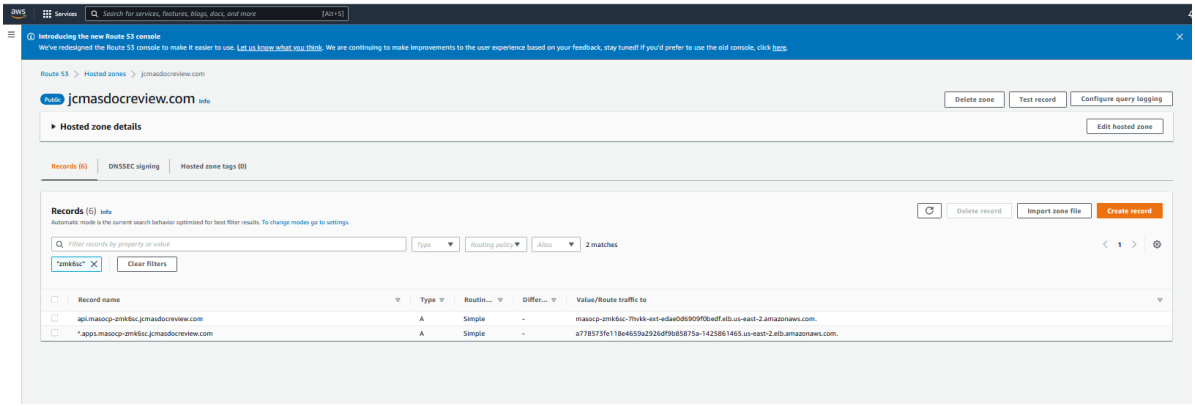
```
{
    "Version": "2012-10-17",
    "Statement": [{
            "Effect": "Allow",
            "Action": "route53:GetChange",
            "Resource": "arn:aws:route53:::change/*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "route53:ChangeResourceRecordSets",
                "route53:ListResourceRecordSets"
            ],
            "Resource": "arn:aws:route53:::hostedzone/*"
        },
        {
            "Effect": "Allow",
            "Action": "route53:ListHostedZonesByName",
            "Resource": "*"
        }
    ]
}
```

   d. Click **Next: Tags**, then **Next: Review**. Provide Name and Description values.
   e. Click **Create policy**.

Procedure

- Amazon Web Services Route 53 configuration
  a) In Route53, go to Hosted Zones, click your domain, and then click **Create record**.

     For example:



Tip: For all examples, replace the parameters given in the example with your own parameters.

b) Add a CNAME record for your Maximo Application Suite instance ID:

   a. Record Name: <mas-instance-id>

   b. Record Value: Load Balancer endpoint, which is located under your Hosted Zones. Filter by your cluster unique ID, and then copy the corresponding value for your cluster ingress.

   Use the second option. For example, the record name beginning with **.apps.masocp-* and the value beginning with *a77* as shown in the screen shot of the previous step.

   Note: Save the instance ID name so you can later use it during the Maximo Application Suite installation.

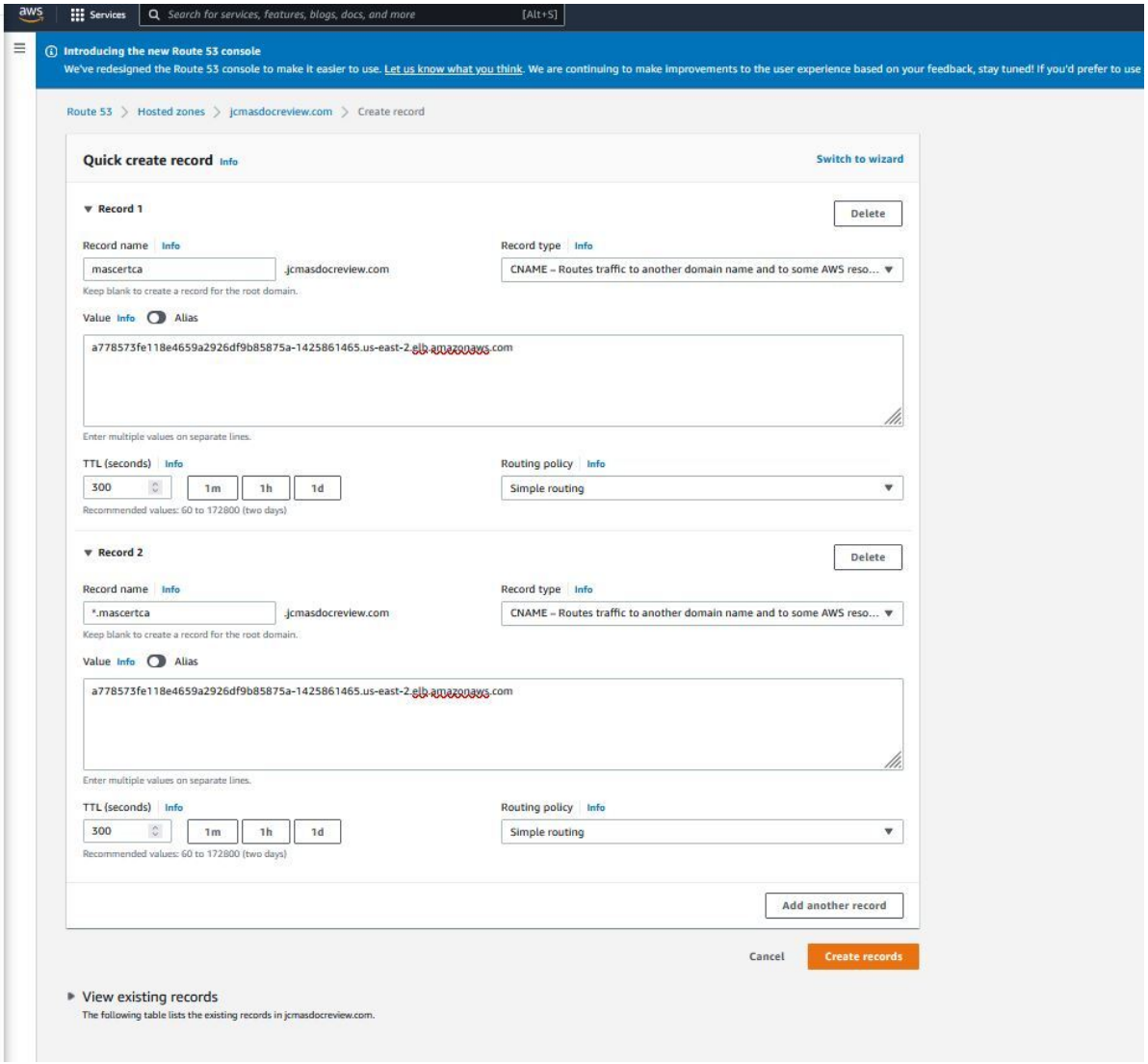
c) Add a new wildcard CNAME record as the following:

   a. Record Name: **.<mas-instance-id>*

   b. Record Value: Load Balancer endpoint, which is found under your Hosted Zones. Filter by your cluster unique ID, and then copy the corresponding value for your cluster ingress.

   Use the second option. For example, the record name beginning with **.apps.masocp-* and the value beginning with *a77* as shown in the screen shot of the previous step.

   For example:



d) Click **Create records**.

- Configure a Let's Encrypt cluster issuer for Maximo Application Suite.

  a) Run the following script in your terminal. You need to be logged into the cluster via oc login command. The script will create a custom cluster issuer named prod-route53-issuer in your cluster.

```
# Export the namespace/project where IBM Certificate Manager is installed in your
cluster. Example: ibm-common-services.
CERT_MANAGER_NAMESPACE=ibm-common-services

# Export your AWS secret access key.
SECRET_ACCESS_KEY=<your aws access key>

# Export your AWS secret access ID.
SECRET_ACCESS_ID=<your aws access id>

# Export your email adress where you'll get alerts and notifications from Let's Encrypt
certificates
EMAIL_ADDRESS=test@test.com

# Export your route53 hosted zone id.
# Find it under AWS console > Route53 > Hosted Zones > search for your Route53 instance,
the hosted zone id will show at the right hand side.
HOSTED_ZONE_ID=<your route53 hosted zone id>

# Create a secret for secret-access-key
oc create secret generic prod-route53-credentials-secret --from-literal=secret-access-
key=${SECRET_ACCESS_KEY} -n ${CERT_MANAGER_NAMESPACE}
```

```
# create a cluster issuer
cat <<EOF > cluster-issuer.yaml
apiVersion: cert-manager.io/v1
kind: ClusterIssuer
metadata:
  name: prod-route53-issuer
spec:
  acme:
    email: ${EMAIL_ADDRESS}
    preferredChain: ''
    privateKeySecretRef:
      name: letsencrypt-prod
    server: 'https://acme-v02.api.letsencrypt.org/directory'
    solvers:
      - dns01:
          route53:
            accessKeyID: ${SECRET_ACCESS_ID}
            hostedZoneID: ${HOSTED_ZONE_ID}
            region: us-east-1
            secretAccessKeySecretRef:
              key: secret-access-key
              name: prod-route53-credentials-secret
EOF
oc apply -f cluster-issuer.yaml -n ${CERT_MANAGER_NAMESPACE}
```

To check if this was properly created, log in to your Red Hat OpenShift cluster, go to **Administration** > **Custom Resource Definitions**, search for **ClusterIssuer** > **Instances**, search for prod-route53-issuer and click it. The cluster issuer shows the following message:

```
The ACME account was registered with the ACME server.
```

b) Select the **Suite** > **YAML** tab and in the **spec** section of the Suite YAML, add **`cluster issue`** and **`domain`** parameters.

```
---
spec:
  certificateIssuer:
    duration: 8760h0m0s
    name: prod-route53-issuer
    renewBefore: 720h0m0s
  domain: <<masinstance_id>>.<<domain>>
```

c) Delete the **`finalizer`** section from the same Suite YAML to force a reconciliation, and then save the YAML file.

```
finalizers:
  - core.mas.ibm.com/finalizer
```

d) In **Networking** under **Routes** of project *mas-<mas_instance_id>-core*, wait for the **Routes** to regenerate for the namespace.

e) Login to the Maximo Application Suite administrator screen and verify the certificate signer.

Note: If IBM Maximo Manage is deployed, the changes might take some time to take effect in Maximo Manage.

- Optional: Configure recursive nameservers

a) On the Red Hat OpenShift web console, select the `ibm-common-services` project.

b) In the **Details** tab of **workloads** > **-> deployment** > **ibm-cert-manager-operator**, scale down the pod from 1 to 0.

c) In the **Deployment** tab, select the `cert-manager controller`.

d) Add the following lines in the yaml file.

```
- '--dns01-recursive-nameservers-only'
- '--dns01-recursive-nameservers=8.8.8.8:53'
```

The following is a sample yaml file.

```
   image: >-
            icr.io/cpopen/cpfs/icp-cert-manager-
controller@sha256:1927c16a4dd369c56fa6d2d1897d3ea3d333a3217b8c05ea32b6617c94833a0e
          args:
            - >-
              --acme-http01-solver-image=icr.io/cpopen/cpfs/icp-cert-manager-
acmesolver@sha256:e8f50ee7b08dc96627e138e9b0d98ed5848c7b4ad92491962c13ef32b2866591
            - '--cluster-resource-namespace=ibm-common-services'
            - '--leader-election-namespace=ibm-common-services'
            - '--dns01-recursive-nameservers-only'
            - '--dns01-recursive-nameservers=8.8.8.8:53'
      serviceAccount: ibm-cert-manager-controller
      dnsPolicy: ClusterFirst
```

Configuring Maximo Application Suite on Amazon Web Services

Use the **Administration** page in Maximo Application Suite to configure the post installation tasks and manage the Maximo Application Suite features.

Setting up Maximo Application Suite environment

The first time that you log in to Maximo Application Suite as the administrator user, no applications are deployed, and no users are logged in to the system. Complete the following tasks to prepare the Maximo Application Suite environment.

About this task

Procedure

In Maximo Application Suite 9.1, select the **Administration** page on side navigation menu from the **Suite** application.

In Maximo Application Suite 9.0 and earlier, select the **Administration** page by clicking the **Administration** icon in the Maximo Application Suite menu bar.

1. On the **Catalog** page, deploy and activate one or more applications.
2. On the **Users** page, add and manage Maximo Application Suite administrators and application users.
3. On the **Configurations** page, administer Maximo Application Suite configurations.
   For example, if you deployed the Maximo Manage application, you can add or update the database configuration information.

   For more information, see Configuring Maximo Application Suite.
4. If you installed Maximo Application Suite core and Maximo Manage, and you want to deploy certain applications and add-ons in Maximo Application Suite, such as Maximo Monitor, you must first install Cloud Pak for Data.

   For more information, see Installing Cloud Pak for Data on an Amazon Web Services instance of Maximo Application Suite.
5. Optional: You can change the authentication to use LDAP or any other supported authentication method.

   The default authentication uses Maximo Application Suite authentication.

   For more information, see Authentication methods.

Accessing the Bootnode and Red Hat OpenShift cluster

By using Secure Shell (SSH) public key authentication, you can access the Bootnode, the bastion host, and the Red Hat OpenShift cluster nodes.

For operational reasons, you might need command-line access to the Bootnode, the bastion host, or the cluster nodes that are located in the virtual private cloud (VPC) of Maximo Application Suite.

About this task

In the AWS cloud, when you start the Maximo Application Suite installation, a Bootnode is created. By using the required tools and the installation parameters, the Bootnode completes the installation.

In the Red Hat OpenShift cluster that is created during the installation, in a public subnet, a bastion host is created. By using this host, you can connect to the cluster nodes in the private subnets.

The Bootnode, bastion host, and private cluster nodes are all Amazon EC2 instances. To maintain or troubleshooting an EC2 instance, connect to it by using Secure Shell (SSH) public key authentication.

Before you installed Maximo Application Suite, you generated a key pair, which consists of a public key and a private key, and uploaded this pair to the Amazon EC2 service. You stored the private key locally. When you specified the installation parameters, you selected the public key in the SSHKey parameter.

During the installation, a copy of the public key is stored in the Bootnode, the bastion host, and the private cluster nodes. Because you have the corresponding private key, you can access these instances by using SSH. In addition, you can use the SSH authentication agent to connect to these instances by using single sign-on authentication.

To use SSH access to connect to the Bootnode, the bastion host, and the private cluster nodes, complete the following steps.

Procedure

1. In your AWS account, connect to the EC2 service console.
2. In the EC2 console, click **Instances**.
3. Retrieve the location of the instance that you want to connect to.
   a) To retrieve the Bootnode details, search for `bootnode`.
   b) To retrieve the bastion host details, search for `bastion-host`. If you want to connect to the private cluster nodes, you must first connect to the bastion host.
   c) Click the instance and copy its location from either the **Public IPV4 address** or **Public IPV4 DNS** fields.
   d) In the **Instance state** column, if the instance is in a shutdown state, click **Start instance**.
4. In your local machine, change the permissions of the private key that you generated before you installed Maximo Application Suite.
   For example, for Linux servers, if you stored the private key in the `/tmp/mas-aws-ssh-key.pem` file, run the following command:

   ```
   chmod 0400 /tmp/mas-aws-ssh-key.pem
   ```

5. If the SSH authentication agent program is not started, run the following command.

   ```
   eval `ssh-agent -s`
   ```

6. Add your private key file into the SSH authentication agent by running the following command:

   ```
   ssh-add -k /tmp/mas-aws-ssh-key.pem
   ```

   You can now connect to the instance by using single sign-on authentication.
7. By using the instance location that you retrieved in “3.c” on page 56, connect to the instance by running the following command:

   ```
   ssh -A ec2-user@<instance-location>
   ```

   For example, to connect to the instance that is at the IP address 35.161.112.157, run the following command:

   ```
   ssh -A ec2-user@35.161.112.157
   ```

8. Optional: If you accessed the bastion host, connect to a private cluster node.

a) Use the Red Hat OpenShift web console to connect to the cluster as an administrator.

b) In the OCP console, click **Home** > **Overview**.

c) In the **Cluster Inventory** card, click the link to the node information.

d) In the **Nodes** page, click the cluster node that you want to connect to.

e) Click **Node details** and record the name of the node.

f) In the bastion host command shell, access the node.

```
ssh core@<node_name>
```

For example, to access the `ip-10-0-132-250.ec2.internal` node, run the following command:

```
ssh core@ip-10-0-132-250.ec2.internal
```

The Bootnode and the bastion host

When you start an IBM Maximo Application Suite installation, a Bootnode is created that controls and completes the installation. In the Red Hat OpenShift cluster, a bastion host is created to allow Secure Shell (SSH) access to cluster nodes.

Note: The bastion host is not created for an existing infrastructure.

During an Maximo Application Suite installation on Amazon Web Services, virtual private clouds (VPC) are created that contain Amazon EC2 instances. For example, a VPC is created to contain the Red Hat OpenShift cluster, and EC2 instances are created in the cluster to represent its master and worker nodes.

The two most important EC2 instances that are created during an installation are the Bootnode and the bastion host.

The Bootnode

In the AWS cloud, after you specify the installation parameters and start the installation, a Bootnode is created. The installation parameters are passed to the Bootnode. In addition, all of the required tools to complete the installation, such as Terraform and Docker, are installed on the Bootnode. By using these tools and parameters, the Bootnode performs the following tasks to complete the installation:

- Creates the virtual network infrastructure, such as the VPC that contains the Red Hat OpenShift cluster.
- Runs the bootstrap process that creates the Red Hat OpenShift cluster.
- Installs the Maximo Application Suite prerequisites.
- Installs the Maximo Application Suite.
- Performs any required postinstallation validation.
- Stores the installation context and Terraform state files both locally and in the Amazon S3 storage bucket that is associated with your AWS account.

Because it is located in its own VPC, the Bootnode is not part of the Maximo Application Suite Red Hat OpenShift cluster. After the installation is complete, you do not need to use the Bootnode to access the cluster or interact with Maximo Application Suite. For this reason, the Bootnode is kept in a shutdown state. However, if required, you can restart it and use it to troubleshoot installation issues.

Note: If you use an existing infrastructure, the Bootnode is in the same VPC of the Red Hat OpenShift Container Platform cluster.

The bastion host

The VPC that the Bootnode creates contains several public and private subnets. In one of the public subnets, a bastion host is created. By using this host, you can connect to the cluster nodes in the private subnets.

After the installation is complete, the bastion host is kept in a shutdown state. However, you can restart it if you want to access cluster nodes by using SSH. For more information, see the Accessing the Bootnode and Red Hat OpenShift cluster topic.

In addition, if required, you can delete the bastion host and create your own.

Note: No charges apply to Amazon EC2 instances that are in a shutdown state, such as the Bootnode and the bastion host. However, charges apply for their attached EBS GP2 volumes of 10 GB. For more information, see Amazon EBS pricing in the AWS documentation.

Managing Red Hat OpenShift cluster on Amazon Web Services

You can shut down nodes in your Red Hat OpenShift cluster that is deployed on Amazon Web Services and restart the EC2 instances.

About this task

Note: To avoid certificate issues, do not restart or shut down your cluster within the first 24 hours after you install IBM Maximo Application Suite.

The official Red Hat OpenShift documentation discusses how to gracefully shut down and restart clusters. Follow the procedure in the Red Hat OpenShift documentation.

The following procedure describes how to shut down nodes in clusters deployed on Amazon Web Services.

Procedure

1. In the Amazon EC2 Dashboard for the Amazon Web Services region where your Red Hat OpenShift cluster is deployed, click **Instances (running)** or **Instances**.

   If you click Instances (running), the resulting list of instances are filtered by `Instance state = running`. If you click Instances, all instances that are stopped or running, appear in the list.

   Instances that are part of the Red Hat OpenShift cluster must have names that have a format `masocp-<unique-string>-<some string>-<nodename>`, where **`unique-string`** is the installation identifier. For more information, see Maximo Application Suite unique identifiers for Amazon Web Services.

   Tip: If the list includes instances that are not related to your Red Hat OpenShift cluster, you can filter by using the **`unique-string`**.
2. Select the nodes that you want to stop or restart by checking the box next to the instance names.
3. Click **Instance state** to stop, restart, or terminate the instance.

   It is recommended that you stop the worker nodes first before you shut down the primary nodes. When you restart a cluster, start the primary nodes before you start the worker nodes.

   Tip: If the cluster is shut down within 24 hours of creation, the following error is shown when you connect to the cluster from a browser or use the **`oc login`** command:

   ```
   error:  Unable to connect to the server: EOF
   ```

   For more information, see How to renew Master node Certificate in Red Hat OpenShift 4.x.

Managing IBM Cloud Pak for Data and IBM Db2 Warehouse

Enable internet access to IBM Db2 Warehouse on a Amazon Web Services Red Hat OpenShift cluster and delete the default Db2 Warehouse database instance that is installed with IBM Cloud Pak for Data.

Procedure

Enable internet access to a Db2 Warehouse on an Amazon Web Services Red Hat OpenShift cluster.

Sometimes it can be helpful to setup internet access to the Db2 Warehouse pods that run in Red Hat OpenShift on Amazon Web Services. Using your database client, you can connect to the database, browse

tables, and run queries. The following task shows how to setup an external route and connect to Db2 Warehouse from a database client on your computer.

1. Get the **NodePort** of the Db2 Warehouse **engn service**.
   For example, 32589.

```
oc get svc -n <cp4d-namespace> | grep engn-svc
```

2. Retrieve the Db2 CA certificate from Db2 CA secret, and save it to a temporary pem file.

```
oc get secret zen-ca-cert-secret -n <cp4d-namespace> -o jsonpath="{.data.ca\.crt}" | base64
-d > /tmp/db2-ca.pem
```

3. Transfer the pem file to jks file and set the password.

```
keytool -import -file /tmp/db2-ca.pem -keystore /tmp/db2-ca-truststore.jks
```

4. Retrieve the IBM Maximo Manage Db2 username and password.

```
oc get secret mas-s58gpv-wsmasocp-jdbc-binding-450f5de9 -n <manage-namespace> -o
jsonpath="{.data.username}" | base64 -d ; echo
```

```
oc get secret mas-s58gpv-wsmasocp-jdbc-binding-450f5de9 -n <manage-namespace> -o
jsonpath="{.data.password}" | base64 -d ; echo
```

5. In Amazon Web Services **Route 53** service, click **Hosted zones** > **<domain name for your cluster>** > **Hosted zones details**, and find the associated VPCs.
6. In **Records**, note the DNS name of the record.
   For example, *.apps.masocp-s58gpv.masawsdoc.com
7. In the Amazon EC2 service, click **Load Balancers**, search the VPC name, and click the load balancer that matches the DNS name.
8. In the **Listeners** tab, add the port to the listener.
9. n the **Description** tab, click the load balancer security group to jump to **Security Groups** page.
10. Select the security group with no name, in the **Inbound rules**, and grant the access to the port.
11. Open your database client tool and configure the connection.
    For example, by using DBeaver v6.2.1, you can configure the following information.

```
Host: <any text>.apps.masocp-s58gpv.masawsdoc.com
Port: 32589
Database: BLUDB:sslConnection=true;sslTrustStoreLocation=/tmp/db2-ca-
truststore.jks;sslTrustStorePassword=123456;
Username: s58gpvmanagedb
Password: s58gpvmanagedbs58gpvmanagedb
JDBC Driver Class Name: com.ibm.db2.jcc.DB2Driver
```

    You can now connect to the Db2 instance from outside the Red Hat OpenShift cluster.

Deleting the default Db2 Warehouse database instance installed with Cloud Pak for Data.

When you deploy the Maximo Application Suite and Cloud Pak for Data deployment option or install Cloud Pak for Data by using the script in your Red Hat OpenShift cluster that is provisioned through the Maximo Application Suite and Manage option of the automated deployment, Cloud Pak for Data is installed with the Db2 Warehouse service and creates a database instance. However, that database instance is not displayed in the Cloud Pak for Data user interface. If you want to delete that instance, going through the Cloud Pak for Data and Databases view, and select the option to delete your database instance, you must delete it by following these steps:

12. In Red Hat OpenShift web console **Administration** > **Custom Resource Definitions**, search for db2ucluster.
13. Click the cluster name and access **Instances**.

14. Click three dots icon for the database instance and delete the instance.

    The `db2ucluster cr` is deleted with resources that are related to the database instance.

    Note: These steps do not uninstall the Db2 Warehouse service on Cloud Pak for Data. For more information about uninstalling the service, see Cloud Pak for Data : Uninstalling Db2 Warehouse.

Configuring the connection to Amazon Web Services DocumentDb

You can configure an Amazon Web Services DocumentDB instance manually from the default MongoDB instance after you install and deploy IBM Maximo Application Suite on Amazon Web Services.

Note: By default, Maximo Application Suite is configured with MongoDB. Therefore, your only option is to configure a new Maximo Application Suite instance to connect with DocumentDB. You cannot migrate your existing data from MongoDB.

Mongocfg and LicenseService CRD

To view the MongoDB configuration details, on the Red Hat OpenShift consoles, select **Administration** > **CustomResourceDefinition** > **MongoCfg CRD** and search for `MongoCfg.config.mas.ibm.com`. Select the **Instances** tab and click the CRD name, for example, **MongoCfg** in bneluv-mongo-system.

To view the MongoDB connection details, select the **YAML** tab.

Similarly, IBM Suite License Service is connected with MongoDB by using the `LicenseService.sls.ibm.com` CRD.

To view the MongoDB configuration details in Suite License Service, on the Red Hat OpenShift console, select **Administration** > **CustomResourceDefinition** > **LicenseService** and search for `LicenseService.sls.ibm.com`. Select the **Instances** tab and click the CRD name, for example, **LicenseService** in masocp-bneluv.

To view the specific variables for DocumentDB that are updated after the DocumentDB instance is provisioned, click the **YAML** tab.

Provisioning Amazon Web Services DocumentDB

You can provision an Amazon Web Services DocumentDB instance after you install and deploy Maximo Application Suite from Amazon Web Services.

Procedure

1. Open the Amazon Web Services console where the Maximo Application Suite is deployed and ensure that a DocumentDB instance is not provisioned.
2. Create a subnet group.
   a) Select a Virtual Private Cloud (VPC) that contains a Maximo Application Suite unique string. For example, select **masocp-bneluv-vpc** from the list.
   b) Add all the subnets that are related to this VPC.
3. Create a cluster parameter group.
   a) Type a new cluster parameter group name.
   b) Select a document database family. For example, select `docdb4.0` from the list.
4. Create a security group.
   a) Select the unique string that was created for the VPC.
   b) For inbound rules, select **Custom TCP** for the rule type and **Custom** for the source.
   c) For outbound rules, select **All traffic** for the rule type and **Anywhere-PIV4** as the destination.
5. On the Amazon Web Services Key Management Service (KMS) console, create a key.
   a) Configure the key type, usage, and advanced options, such as key material origin and region.

b) Add labels, such as the alias, description, and tags.

c) Configure the key usage details, such as key administrators and permissions.

d) Review your key policy and click **finish**.

6. Create a DocumentDB cluster.

a) Select **5.0.0** for the engine version to create the cluster.

Note: Maximo Application Suite supports only DocumentDB 5.0.0.

b) Set **Show Advanced settings** to on.

c) In Network Setting section, select the VPC that contains the unique string for Maximo Application Suite.

The subnet group is selected automatically when you select the VPC.

d) Select the security group that was created earlier. Remove any security group that is selected by default.

e) Under cluster options, select the parameter group that was created earlier.

f) Under encryption options, enable encryption and choose the KMS key that was created earlier.

g) Create the DocumentDB cluster with other default options, such as maintenance, tags, and deletion protection.

The cluster is deployed in some time, for example, in 15 minutes.

7. Optional: Verify the DocumentDB connection by using a DB client.
For example you can use Studio 3T to verify the connection. For more information, see Connecting to an Amazon DocumentDB cluster from Studio 3T.

What to do next

Configure a Maximo Application Suite instance to connect with the Amazon Web Services DocumentDB instance, update the `MongoCfg CRD` and `LicenseService CRD` with the Amazon Web Services DocumentDB connection details.

For more information, see "Configuring Maximo Application Suite with Amazon Web Services DocumentDB" on page 61

Configuring Maximo Application Suite with Amazon Web Services DocumentDB

To configure a new Maximo Application Suite instance to connect with the Amazon Web Services DocumentDB instance, on the Red Hat OpenShift console, update the MongoCfg CRD and LicenseService CRD with the Amazon Web Services DocumentDB connection details.

About this task

Note: You can configure a new Maximo Application Suite instance to connect with DocumentDB by using the procedure that is described in this document. However, you cannot migrate your existing data from MongoDB by using this procedure.

Procedure

1. Log in to the Red Hat OpenShift console as an admin user.
2. From the side navigation menu, select **Administration** > **CustomResourceDefinition** and in the list, click MongoCfg CRD.
3. Click **Instances** tab.
4. Update the MongoCfg CRD with the Amazon Web Services DocumentDB connection details.

By default Maximo Application Suite uses the MongoCfg CRD to connect to MongoDB. To connect to the DocumentDB, you must update the connection details on the **YAML** tab on the Red Hat OpenShift console.

a) Create a backup file of the MongoCfg CRD YAML file.

To create a backup file, either copy and paste the MongoCfg CRD YAML file contents to a file or use the following **oc** command:

```
oc get MongoCfg <instance name> -o yaml > instance_name.yaml
```

b) Create a secret *docdb-dbadmin* YAML file in the `mas-<ClusterUniqueString>-core` namespace.
The following example is a sample secret YAML file in the mas-vbpoyr-core namespace.

```
---
# DocumentDB credentials for Core
apiVersion: v1
kind: Secret
type: Opaque
metadata:
  name: documentdb-admin
  namespace: mas-vbpoyr-core
stringData:
  username: docdbadmin
  password: docdbadmin
```

c) Update the MongCfg CRD YAML file with the DocumentDB connection details.

- **spec.certificates** – The certificate for the specific DocumentDB. Amazon Web Services DocumentDB provides a PEM file.

  If the certificate authority is `rds-ca-2019`, then download the PEM files from Certificate bundles for specific AWS Regions. For example, for ca-central-1 region, you can use the PEM file from ca-central-1-bundle.pem.

  However, if the certificate authority is `rds-ca-rsa2048-g1`, then to download the `Root CA RSA2048 PEM` files follow steps provided in the Extracting `Root CA RSA2048 PEM` files section.
- **spec.config.credentials.secretName** – The secret that contains the DocumentDB admin username and password.
- **spec.config.hosts.host** – DocumentDB hostname.
- **spec.config.hosts.port** – DocumentDB port.
- **spec.config.retryWrites** – Set to false.

The following example is a sample MongoCfg CRD YAML file that is updated with DocumentDB connection details:

```
---
apiVersion: config.mas.ibm.com/v1
kind: MongoCfg
metadata:
  resourceVersion: '369530'
  name: vbpoyr-mongo-system
  uid: 48318d43-b0f6-4848-822b-4443a104b8f0
  generation: 1
  namespace: mas-vbpoyr-core
  ownerReferences:
    - apiVersion: core.mas.ibm.com/v1
      kind: Suite
      name: vbpoyr
      uid: 0df07123-1ad0-492a-aec0-76f88583dd98
  labels:
    mas.ibm.com/configScope: system
    mas.ibm.com/instanceId: vbpoyr
spec:
  certificates:
    - alias: ca
      crt: |
        -----BEGIN CERTIFICATE-----
        MIIEBjCCAu6gAwIBAgIJAMc0ZzaSUK51MA0GCSqGSIb3DQEBCwUAMIGPMQswCQYD
        VQQGEwJVUzEQMA4GA1UEBwwHU2VhdHRsZTETMBEGA1UECAwKV2FzaGluZ3RvbjEi
        MCAGA1UECgwZQW1hem9uIFdlYiBTZXJ2aWNlcywgSW5jLjETMBEGA1UECwwKQW1h
        em9uIFJEUzEgMB4GA1UEAwwXQW1hem9uIFJEUyBSb290IDIwMTkgQ0EwHhcNMTkw
        ODIyMTcwODUwWhcNMjQwODIyMTcwODUwWjCBjzELMAkGA1UEBhMCVVMxEDAOBgNV
        BAcMB1NlYXR0bGUxEzARBgNVBAgMCldhc2hpbmd0b24xIjAgBgNVBAoMGUFtYXpv
        biBXZWIgU2VydmljZXMsIEluYy4xEzARBgNVBAsMCkFtYXpvbiBSRFMxIDAeBgNV
```

```
BAMMF0FtYXpvbiBSRFMgUm9vdCAyMDE5IENBMIIBIjANBgkqhkiG9w0BAQEFAAOC
AQ8AMIIBCgKCAQEArXnF/E6/Qh+ku3hQTSKPMhQQlCpoWvnIthzX6MK3p5a0eXKZ
oWIjYcNNG6UwJjp4fUXl6glp53Jobn+tWNX88dNH2n8DVbppSwScVE2LpuL+94vY
0EYE/XxN7svKea8YvlrqkUBKyxLxTjh+U/KrGOaHxz9v0l6ZNlDbuaZw3qIWdD/I
6aNbGeRUVtpM6P+bWIoxVl/caQylQS6CEYUk+CpVyJSkopwJlzXT07tMoDL5WgX9
O08KVgDNz9qP/IGtAcRduRcNioH3E9v981QO1zt/Gpb2f8NqAjUUCUZzOnij6mx9
McZ+9cWX88CRzR0vQODWuZscgI08NvM69Fn2SQIDAQABo2MwYTAOBgNVHQ8BAf8E
BAMCAQYwDwYDVR0TAQH/BAUwAwEB/zAdBgNVHQ4EFgQUc19g2LzLA5j0Kxc0LjZa
pmD/vB8wHwYDVR0jBBgwFoAUc19g2LzLA5j0Kxc0LjZapmD/vB8wDQYJKoZIhvcN
AQELBQADggEBAHAG7WTmyjzPRIM85rVj+fWHsLIvqpw6DObIjMWokpliCeMINZFV
ynfgBKsf1ExwbvJNzYFXW6dihnguDG9VMPpi2up/ctQTN8tm9nDKOy08uNZoofMc
NUZxKCEkVKZv+IL4oHoeayt8egtv3ujJM6V14AstMQ6SwvwvA93EP/Ug2e4WAXHu
cbI1NAbUgVDqp+DRdfvZkgYKryjTWd/0+1fS8X1bBZVWzl7eirNVnHbSH2ZDpNuY
0SBd8dj5F6ld3t58ydZbrTHze7JJOd8ijySAp4/kiu9UfZWuTPABzDa/DSdz9Dk/
zPW4CXXvhLmE02TA9/HeCw3KEHIwicNuEfw=
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIECjCCAvKgAwIBAgICEzUwDQYJKoZIhvcNAQELBQAwgY8xCzAJBgNVBAYTAlVT
MRAwDgYDVQQHDAdTZWF0dGxlMRMwEQYDVQQIDApXYXNoaW5ndG9uMSIwIAYDVQQK
DBlBbWF6b24gV2ViIFNlcnZpY2VzLCBJbmMuMRMwEQYDVQQLDApBbWF6b24gUkRT
MSAwHgYDVQQDDBdBbWF6b24gUkRTIFJvb3QgMjAxOSBDQTAeFw0xOTA5MTAyMDUy
MjVaFw0yNDA4MjIxNzA4NTBaMIGXMQswCQYDVQQGEwJVUzETMBEGA1UECAwKV2Fz
aGluZ3RvbjEQMA4GA1UEBwwHU2VhdHRsZTEiMCAGA1UECgwZQW1hem9uIFdlYiBT
ZXJ2aWNlcywgSW5jLjETMBEGA1UECwwKQW1hem9uIFJEUzEoMCYGA1UEAwwfQW1h
em9uIFJEUyBjYS1jZW50cmFsLTEgMjAxOSBDQTCCASIwDQYJKoZIhvcNAQEBBQAD
ggEPADCCAQoCggEBAOxHqdcPSA2uBjsCP4DLSlqSoPuQ/X1kkJLusVRKiQE2zayB
viuCBt4VB9Qsh2rW3iYGM+usDjltGnI1iUWA5KHcvHszSMkWAOYWLiMNKTlg6LCp
XnE89tvj5dIH6U8WlDvXLdjB/h30gW9JEX7S8supsBSci2GxEzb5mRdKaDuuF/0O
qvz4YE04pua3iZ9QwmMFuTAOYzD1M72aOpj+7Ac+YLMM61qOtU+AU6MndnQkKoQi
qmUN2A9IFaqHFzRlSdXwKCKUA4otzmz+/N3vFwjb5F4DSsbsrMfjeHMo6o/nb6Nh
YDb0VJxxPee6TxSuN7CQJ2FxMlFUezcoXqwqXD0CAwEAAaNmMGQwDgYDVR0PAQH/
BAQDAgEGMBIGA1UdEwEB/wQIMAYBAf8CAQAwHQYDVR0OBBYEFDGGpon9WfIpsggE
CxHq8hZ7E2ESMB8GA1UdIwQYMBaAFHNfYNi8ywOY9CsXNC42WqZg/7wfMA0GCSqG
SIb3DQEBCwUAA4IBAQAvpeQYEGZvoTVLgV9rd2+StPYykMsmFjWQcyn3dBTZRXC2
lKq7QhQczMAOhEaaN29ZprjQzsA2X/UauKzLR2Uyqc2qOeO9/YOl0H3qauo8C/W9
r8xqPbOCDLEXlOQ19fidXyyEPHEq5WFp8j+fTh+s8WOx2M7IuC0ANEetIZURYhSp
xl9XOPRCJxOhj7JdelhpweX0BJDNHeUFi0ClnFOws8oKQ7sQEv66d5ddxqqZ3NVv
RbCvCtEutQMOUMIuaygDlMn1anSM8N7Wndx8G6+Uy67AnhjGx7jw/0YPPxopEj6x
JXP8j0sJbcT9K/9/fPVLNT25RvQ/93T2+IQL4Ca2
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIICtTCCAjqgAwIBAgIRAK9BSZU6nIe6jqfODmuVctYwCgYIKoZIzj0EAwMwgZkx
CzAJBgNVBAYTAlVTMSIwIAYDVQQKDBlBbWF6b24gV2ViIFNlcnZpY2VzLCBJbmMu
MRMwEQYDVQQLDApBbWF6b24gUkRTMQswCQYDVQQIDAJXQTEyMDAGA1UEAwwpQW1h
em9uIFJEUyBjYS1jZW50cmFsLTEgUm9vdCBDQSBFQ0MzODQgRzExEDAOBgNVBAcM
B1NlYXR0bGUwIBcNMjEwNTIxMjIxMzA5WhgPMjEyMTA1MjEyMzEzMDlaMIGZMQsw
CQYDVQQGEwJVUzEiMCAGA1UECgwZQW1hem9uIFdlYiBTZXJ2aWNlcywgSW5jLjET
MBEGA1UECwwKQW1hem9uIFJEUzELMAkGA1UECAwCV0ExMjAwBgNVBAMMKUFtYXpv
biBSRFMgY2EtY2VudHJhbC0xIFJvb3QgQ0EgRUNDMzg0IEcxMRAwDgYDVQQHDAdT
ZWF0dGxlMHYwEAYHKoZIzj0CAQYFK4EEACIDYgAEUkEERcgxneT5H+P+fERcbGmf
bVx+M7rNWtgWUr6w+OBENebQA9ozTkeSg4c4M+qdYSObFqjxITdYxT1z/nHz1gyx
OKAhLjWu+nkbRefqy3RwXaWT680uUaAP6ccnkZOMo0IwQDAPBgNVHRMBAf8EBTAD
AQH/MB0GA1UdDgQWBBSN6fxlg0s5Wny08uRBYZcQ3TUoyzAOBgNVHQ8BAf8EBAMC
AYYwCgYIKoZIzj0EAwMDaQAwZgIxAORaz+MBVoFBTmZ93j2G2vYTwA6T5hWzBWrx
CrI54pKn5g6At56DBrkjrwZF5T1enAIxAJe/LZ9xpDkAdxDgGJFN8gZYLRWc0NRy
Rb4hihy5vj9L+w9uKc9VfEBIFuhT7Z3ljg==
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIEBTCCAu2gAwIBAgIRAKiaRZatN8eiz9p0s0lu0rQwDQYJKoZIhvcNAQELBQAw
gZoxCzAJBgNVBAYTAlVTMSIwIAYDVQQKDBlBbWF6b24gV2ViIFNlcnZpY2VzLCBJ
bmMuMRMwEQYDVQQLDApBbWF6b24gUkRTMQswCQYDVQQIDAJXQTEzMDEGA1UEAwwq
QW1hem9uIFJEUyBjYS1jZW50cmFsLTEgUm9vdCBDQSBSU0EyMDQ4IEcxMRAwDgYD
VQQHDAdTZWF0dGxlMCAXDTIxMDUyMTIyMDIzNVoYDzIwNjEwNTIxMjMwMjM1WjCB
mjELMAkGA1UEBhMCVVMxIjAgBgNVBAoMGUFtYXpvbiBXZWIgU2VydmljZXMsIElu
Yy4xEzARBgNVBAsMCkFtYXpvbiBSRFMxCzAJBgNVBAgMAldBMTMwMQYDVQQDDCpB
bWF6b24gUkRTIGNhLWNlbnRyYWwtMSBSb290IENBIFJTQTIwNDggRzExEDAOBgNV
BAcMB1NlYXR0bGUwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQCygVMf
qB865IR9qYRBRFHn4eAqGJOCFx+UbraQZmjr/mnRqSkY+nhbM7Pn/DWOrRnxoh+w
q5F9ZxdZ5D5T1v6kljVwxyfFgHItyyyIL0YS7e2h7cRRscCM+75kMedAP7icb4YN
LfWBqfKHbHIOqvvQK8T6+Emu/QlG2B5LvuErrop9K0KinhITekpVIO4HCN61cuOe
CADBKF/5uUJHwS9pWw3uUbpGUwsLBuhJzCY/OpJlDqC8Y9aToi2Ivl5u3/Q/sKjr
6AZb9lx4q3J2z7tJDrm5MHYwV74elGSXoeoG8nODUqjgklIWAPrt6lQ3WJpO2kug
8RhCdSbWkcXHfX95AgMBAAGjQjBAMA8GA1UdEwEB/wQFMAMBAf8wHQYDVR0OBBYE
FOIxhqTPkKVqKBZvMWtKewKWDvDBMA4GA1UdDwEB/wQEAwIBhjANBgkqhkiG9w0B
AQsFAAOCAQEAqoItII89lOl4TKvg0I1EinxafZLXIheLcdGCxpjRxlZ9QMQUN3yb
y/8uFKBL0otbQgJEoGhxm4h0tp54g28M6TN1U0332dwkjYxUNwvzrMaV5Na55I2Z
1hq4GB3NMXW+PvdtsgVOZbEN+zOyOZ5MvJHEQVkT3YRnf6avsdntltcRzHJ16pJc
Y8rR7yWwPXh1lPaPkxddrCtwayyGxNbNmRybjR48uHRhwu7v2WuAMdChL8H8bp89
TQLMrMHgSbZfee9hKhO4Zebelf1/cslRSrhkG0ESq6G5MUINj6lMg2g6F0F7Xz2v
ncD/vuRN5P+vT8th/oZ0Q2Gc68Pun0cn/g==
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
```

```
        MIIGBTCCA+2gAwIBAgIRAJfKe4Zh4aWNt3bv6ZjQwogwDQYJKoZIhvcNAQEMBQAw
        gZoxCzAJBgNVBAYTAlVTMSIwIAYDVQQKDBlBbWF6b24gV2ViIFNlcnZpY2VzLCBJ
        bmMuMRMwEQYDVQQLDApBbWF6b24gUkRTMQswCQYDVQQIDAJXQTEzMDEGA1UEAwwq
        QW1hem9uIFJEUyBjYS1jZW50cmFsLTEgUm9vdCBDQSBSU0E0MDk2IEcxMRAwDgYD
        VQQHDAdTZWF0dGxlMCAXDTIxMDUyMTIyMDg1M1oYDzIxMjEwNTIxMjMwODUzWjCB
        mjELMAkGA1UEBhMCVVMxIjAgBgNVBAoMGUFtYXpvbiBXZWIgU2VydmljZXMsIElu
        Yy4xEzARBgNVBAsMCkFtYXpvbiBSRFMxCzAJBgNVBAgMAldBMTMwMQYDVQQDDCpB
        bWF6b24gUkRTIGNhLWNlbnRyYWwtMSBSb290IENBIFJTQTQwOTYgRzExEDAOBgNV
        BAcMB1NlYXR0bGUwggIiMA0GCSqGSIb3DQEBAQUAA4ICDwAwggIKAoICAQCpgUH6
        Crzd8cOw9prAh2rkQqAOx2vtuI7xX4tmBG4I/um28eBjyVmgwQ1fpq0Zg2nCKS54
        Nn0pCmT7f3h6Bvopxn0J45AzXEtajFqXf92NQ3iPth95GVfAJSD7gk2LWMhpmID9
        JGQyoGuDPg+hYyr292X6d0madzEktVVGO4mKTF989qEg+tY8+oN0U2fRTrqa2tZp
        iYsmg350ynNopvntsJAfpCO/srwpsqHHLNFZ9jvhTU8uW90wgaKO9i31j/mHggCE
        +CAOaJCM3g+L8DPl/2QKsb6UkBgaaIwKyRgKSj1IlgrK+OdCBCOgM9jjId4Tqo2j
        ZIrrPBGl6fbn1+etZX+2/tf6tegz+yV0HHQRAcKCpaH8AXF44bny9andslBoNjGx
        H6R/3ib4FhPrnBMElzZ5i4+eM/cuPC2huZMBXb/jKgRC/QN1Wm3/nah5FWq+yn+N
        tiAF10Ga0BYzVhHDEwZzN7gn38bcY5yi/CjDUNpY0OzEe2+dpaBKPlXTaFfn9Nba
        CBmXPRF0lLGGtPeTAgjcju+NEcVa82Ht1pqxyu2sDtbu3J5bxp4RKtj+ShwN8nut
        Tkf5Ea9rSmHEY13fzgibZlQhXaiFSKA2ASUwgJP19Putm0XKlBCNSGCoECemewxL
        +7Y8FszS4Uu4eaIwvXVqUEE2yf+4ex0hqQ1acQIDAQABo0IwQDAPBgNVHRMBAf8E
        BTADAQH/MB0GA1UdDgQWBBSeUnXIRxNbYsZLtKomIz4Y1nOZEzAOBgNVHQ8BAf8E
        BAMCAYYwDQYJKoZIhvcNAQEMBQADggIBAIpRvxVS0dzoosBh/qw65ghPUGSbP2D4
        dm6oYCv5g/zJr4fR7NzEbHOXX5aOQnHbQL4M/7veuOCLNPOW1uXwywMg6gY+dbKe
        YtPVA1as8G9sUyadeXyGh2uXGsziMFXyaESwiAXZyiYyKChS3+g26/7jwECFo5vC
        XGhWpIO7Hp35Yglp8AnwnEAo/PnuXgyt2nvyTSrxlEYa0jus6GZEZd77pa82U1JH
        qFhIgmKPWWdvELA3+ra1nKnvpWM/xX0pnMznMej5B3RT3Y+k61+kWghJE81Ix78T
        +tG4jSotgbaL53BhtQWBD1yzbbilqsGE1/DXPXzHVf9yD73fwh2tGWSaVInKYinr
        a4tcrB3KDN/PFq0/w5/21lpZjVFyu/eiPj6DmWDuHW73XnRwZpHo/2OFkei5R7cT
        rn/YdDD6c1dYtSw5YNnS6hdCQ3sOiB/xbPRN9VWJa6se79uZ9NLz6RMOr73DNnb2
        bhIR9Gf7XAA5lYKqQk+A+stoKbIT0F65RnkxrXi/6vSiXfCh/bV6B41cf7MY/6YW
        ehserSdjhQamv35rTFdM+foJwUKz1QN9n9KZhPxeRmwqPitAV79PloksOnX25ElN
        SlyxdndIoA1wia1HRd26EFm2pqfZ2vtD2EjU3wD42CXX4H8fKVDna30nNFSYF0yn
        jGKc3k6UNxpg
        -----END CERTIFICATE-----
  config:
    authMechanism: DEFAULT
    configDb: admin
    retryWrites: false
    credentials:
      secretName: documentdb-admin
    hosts:
      - host: docdb-vbpoyr.ctnrnscupeqf.ca-central-1.docdb.amazonaws.com
        port: 27017
  displayName: Document Db in 'mongoce-vbpoyr' namespace
  type: external
```

d) Save the MongoCfg CRD YAML file.
After you apply the changes, reconciliation runs to connect to the new DocumentDB instance. On successful connection, the last reconciliation in the condition section is shown as successful.

5. Update the LicenseService CRD with the Amazon Web Services DocumentDB connection details.

a) Create a backup file of the LicenseService Suite License Service service instance YAML file.

b) Create a secret `docdb-dbadmin` YAML file in the `ibm-sls-<ClusterUniqueString>` namespace.
The following example is a sample secret YAML file in the ibm-sls-vbpoyr namespace.

```
---
# DocumentDB credentials for SLS
apiVersion: v1
kind: Secret
type: Opaque
metadata:
  name: documentdb
  namespace: ibm-sls-vbpoyr
stringData:
  username: docdbadmin
  password: docdbadmin
```

c) On the Red Hat OpenShift console, from the side navigation menu, select **CustomResourceDefinition** > **LicenseService**.

d) Select the **Instances** tab and click the CRD name. Update the LicenseService CRD YAML file with the following DocumentDB connection details.

i) **`spec.mongo.certificates`** – The certificate for the specific DocumentDB.

If the certificate authority is `rds-ca-2019`, then download the PEM files from Certificate bundles for specific AWS Regions. For example, for ca-central-1 region, you can use the PEM file from ca-central-1-bundle.pem.

However, if the certificate authority is `rds-ca-rsa2048-g1`, then to download the `Root CA RSA2048 PEM` files follow steps provided in the Extracting `Root CA RSA2048 PEM` files section.

ii) **`spec.mongo.secretName`** – The secret that contains the DocumentDB admin username and password.

iii) **`spec.mongo.nodes.host`** – DocumentDB hostname.

iv) **`spec.mongo.nodes.port`** – DocumentDB port.

v) **`spec.mongo.retryWrites`** – Set to false.

The following example is a sample `LicenseService` YAML file that is updated with DocumentDB connection details:

```
---
apiVersion: sls.ibm.com/v1
kind: LicenseService
metadata:
  name: masocp-vbpoyr
  namespace: ibm-sls-vbpoyr
  resourceVersion: '497850'
  uid: 478dcfb9-342e-440c-b6ec-7ad74b5b78ec
spec:
  domain: ibm-sls-vbpoyr.apps.masocp-vbpoyr.buyermas4aws.com
  license:
    accept: true
  mongo:
    certificates:
      - alias: ca
        crt: |
          -----BEGIN CERTIFICATE-----
          MIIEBjCCAu6gAwIBAgIJAMc0ZzaSUK51MA0GCSqGSIb3DQEBCwUAMIGPMQswCQYD
          VQQGEwJVUzEQMA4GA1UEBwwHU2VhdHRsZTETMBEGA1UECAwKV2FzaGluZ3RvbjEi
          MCAGA1UECgwZQW1hem9uIFdlYiBTZXJ2aWNlcywgSW5jLjETMBEGA1UECwwKQW1h
          em9uIFJEUzEgMB4GA1UEAwwXQW1hem9uIFJEUyBSb290IDIwMTkgQ0EwHhcNMTkw
          ODIyMTcwODUwWhcNMjQwODIyMTcwODUwWjCBjzELMAkGA1UEBhMCVVMxEDAOBgNV
          BAcMB1NlYXR0bGUxEzARBgNVBAgMCldhc2hpbmd0b24xIjAgBgNVBAoMGUFtYXpv
          biBXZWIgU2VydmljZXMsIEluYy4xEzARBgNVBAsMCkFtYXpvbiBSRFMxIDAeBgNV
          BAMMF0FtYXpvbiBSRFMgUm9vdCAyMDE5IENBMIIBIjANBgkqhkiG9w0BAQEFAAOC
          AQ8AMIIBCgKCAQEArXnF/E6/Qh+ku3hQTSKPMhQQlCpoWvnIthzX6MK3p5a0eXKZ
          oWIjYcNNG6UwJjp4fUXl6glp53Jobn+tWNX88dNH2n8DVbppSwScVE2LpuL+94vY
          0EYE/XxN7svKea8YvlrqkUBKyxLxTjh+U/KrGOaHxz9v0l6ZNlDbuaZw3qIWdD/I
          6aNbGeRUVtpM6P+bWIoxVl/caQylQS6CEYUk+CpVyJSkopwJlzXT07tMoDL5WgX9
          O08KVgDNz9qP/IGtAcRduRcNioH3E9v981QO1zt/Gpb2f8NqAjUUCUZzOnij6mx9
          McZ+9cWX88CRzR0vQODWuZscgI08NvM69Fn2SQIDAQABo2MwYTAOBgNVHQ8BAf8E
          BAMCAQYwDwYDVR0TAQH/BAUwAwEB/zAdBgNVHQ4EFgQUc19g2LzLA5j0Kxc0LjZa
          pmD/vB8wHwYDVR0jBBgwFoAUc19g2LzLA5j0Kxc0LjZapmD/vB8wDQYJKoZIhvcN
          AQELBQADggEBAHAG7WTmyjzPRIM85rVj+fWHsLIvqpw6DObIjMWokpliCeMINZFV
          ynfgBKsf1ExwbvJNzYFXW6dihnguDG9VMPpi2up/ctQTN8tm9nDKOy08uNZoofMc
          NUZxKCEkVKZv+IL4oHoeayt8egtv3ujJM6V14AstMQ6SwvwvA93EP/Ug2e4WAXHu
          cbI1NAbUgVDqp+DRdfvZkgYKryjTWd/0+1fS8X1bBZVWzl7eirNVnHbSH2ZDpNuY
          0SBd8dj5F6ld3t58ydZbrTHze7JJOd8ijySAp4/kiu9UfZWuTPABzDa/DSdz9Dk/
          zPW4CXXvhLmE02TA9/HeCw3KEHIwicNuEfw=
          -----END CERTIFICATE-----
          -----BEGIN CERTIFICATE-----
          MIIECjCCAvKgAwIBAgICEzUwDQYJKoZIhvcNAQELBQAwgY8xCzAJBgNVBAYTAlVT
          MRAwDgYDVQQHDAdTZWF0dGxlMRMwEQYDVQQIDApXYXNoaW5ndG9uMSIwIAYDVQQK
          DBlBbWF6b24gV2ViIFNlcnZpY2VzLCBJbmMuMRMwEQYDVQQLDApBbWF6b24gUkRT
          MSAwHgYDVQQDDBdBbWF6b24gUkRTIFJvb3QgMjAxOSBDQTAeFw0xOTA5MTAyMDUy
          MjVaFw0yNDA4MjIxNzA4NTBaMIGXMQswCQYDVQQGEwJVUzETMBEGA1UECAwKV2Fz
          aGluZ3RvbjEQMA4GA1UEBwwHU2VhdHRsZTEiMCAGA1UECgwZQW1hem9uIFdlYiBT
          ZXJ2aWNlcywgSW5jLjETMBEGA1UECwwKQW1hem9uIFJEUzEoMCYGA1UEAwwfQW1h
          em9uIFJEUyBjYS1jZW50cmFsLTEgMjAxOSBDQTCCASIwDQYJKoZIhvcNAQEBBQAD
          ggEPADCCAQoCggEBAOxHqdcPSA2uBjsCP4DLSlqSoPuQ/X1kkJLusVRKiQE2zayB
          viuCBt4VB9Qsh2rW3iYGM+usDjltGnI1iUWA5KHcvHszSMkWAOYWLiMNKTlg6LCp
          XnE89tvj5dIH6U8WlDvXLdjB/h30gW9JEX7S8supsBSci2GxEzb5mRdKaDuuF/0O
          qvz4YE04pua3iZ9QwmMFuTAOYzD1M72aOpj+7Ac+YLMM61qOtU+AU6MndnQkKoQi
          qmUN2A9IFaqHFzRlSdXwKCKUA4otzmz+/N3vFwjb5F4DSsbsrMfjeHMo6o/nb6Nh
          YDb0VJxxPee6TxSuN7CQJ2FxMlFUezcoXqwqXD0CAwEAAaNmMGQwDgYDVR0PAQH/
          BAQDAgEGMBIGA1UdEwEB/wQIMAYBAf8CAQAwHQYDVR0OBBYEFDGGpon9WfIpsggE
          CxHq8hZ7E2ESMB8GA1UdIwQYMBaAFHNfYNi8ywOY9CsXNC42WqZg/7wfMA0GCSqG
          SIb3DQEBCwUAA4IBAQAvpeQYEGZvoTVLgV9rd2+StPYykMsmFjWQcyn3dBTZRXC2
          lKq7QhQczMAOhEaaN29ZprjQzsA2X/UauKzLR2Uyqc2qOeO9/YOl0H3qauo8C/W9
```

```
        r8xqPbOCDLEXlOQ19fidXyyEPHEq5WFp8j+fTh+s8WOx2M7IuC0ANEetIZURYhSp
        xl9XOPRCJxOhj7JdelhpweX0BJDNHeUFi0ClnFOws8oKQ7sQEv66d5ddxqqZ3NVv
        RbCvCtEutQMOUMIuaygDlMn1anSM8N7Wndx8G6+Uy67AnhjGx7jw/0YPPxopEj6x
        JXP8j0sJbcT9K/9/fPVLNT25RvQ/93T2+IQL4Ca2
        -----END CERTIFICATE-----
        -----BEGIN CERTIFICATE-----
        MIICtTCCAjqgAwIBAgIRAK9BSZU6nIe6jqfODmuVctYwCgYIKoZIzj0EAwMwgZkx
        CzAJBgNVBAYTAlVTMSIwIAYDVQQKDBlBbWF6b24gV2ViIFNlcnZpY2VzLCBJbmMu
        MRMwEQYDVQQLDApBbWF6b24gUkRTMQswCQYDVQQIDAJXQTEyMDAGA1UEAwwpQW1h
        em9uIFJEUyBjYS1jZW50cmFsLTEgUm9vdCBDQSBFQ0MzODQgRzExEDAOBgNVBAcM
        B1NlYXR0bGUwIBcNMjEwNTIxMjIxMzA5WhgPMjEyMTA1MjEyMzEzMDlaMIGZMQsw
        CQYDVQQGEwJVUzEiMCAGA1UECgwZQW1hem9uIFdlYiBTZXJ2aWNlcywgSW5jLjET
        MBEGA1UECwwKQW1hem9uIFJEUzELMAkGA1UECAwCV0ExMjAwBgNVBAMMKUFtYXpv
        biBSRFMgY2EtY2VudHJhbC0xIFJvb3QgQ0EgRUNDMzg0IEcxMRAwDgYDVQQHDAdT
        ZWF0dGxlMHYwEAYHKoZIzj0CAQYFK4EEACIDYgAEUkEERcgxneT5H+P+fERcbGmf
        bVx+M7rNWtgWUr6w+OBENebQA9ozTkeSg4c4M+qdYSObFqjxITdYxT1z/nHz1gyx
        OKAhLjWu+nkbRefqy3RwXaWT680uUaAP6ccnkZOMo0IwQDAPBgNVHRMBAf8EBTAD
        AQH/MB0GA1UdDgQWBBSN6fxlg0s5Wny08uRBYZcQ3TUoyzAOBgNVHQ8BAf8EBAMC
        AYYwCgYIKoZIzj0EAwMDaQAwZgIxAORaz+MBVoFBTmZ93j2G2vYTwA6T5hWzBWrx
        CrI54pKn5g6At56DBrkjrwZF5T1enAIxAJe/LZ9xpDkAdxDgGJFN8gZYLRWc0NRy
        Rb4hihy5vj9L+w9uKc9VfEBIFuhT7Z3ljg==
        -----END CERTIFICATE-----
        -----BEGIN CERTIFICATE-----
        MIIEBTCCAu2gAwIBAgIRAKiaRZatN8eiz9p0s0lu0rQwDQYJKoZIhvcNAQELBQAw
        gZoxCzAJBgNVBAYTAlVTMSIwIAYDVQQKDBlBbWF6b24gV2ViIFNlcnZpY2VzLCBJ
        bmMuMRMwEQYDVQQLDApBbWF6b24gUkRTMQswCQYDVQQIDAJXQTEzMDEGA1UEAwwq
        QW1hem9uIFJEUyBjYS1jZW50cmFsLTEgUm9vdCBDQSBSU0EyMDQ4IEcxMRAwDgYD
        VQQHDAdTZWF0dGxlMCAXDTIxMDUyMTIyMDIzNVoYDzIwNjEwNTIxMjMwMjM1WjCB
        mjELMAkGA1UEBhMCVVMxIjAgBgNVBAoMGUFtYXpvbiBXZWIgU2VydmljZXMsIElu
        Yy4xEzARBgNVBAsMCkFtYXpvbiBSRFMxCzAJBgNVBAgMAldBMTMwMQYDVQQDDCpB
        bWF6b24gUkRTIGNhLWNlbnRyYWwtMSBSb290IENBIFJTQTIwNDggRzExEDAOBgNV
        BAcMB1NlYXR0bGUwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQCygVMf
        qB865IR9qYRBRFHn4eAqGJOCFx+UbraQZmjr/mnRqSkY+nhbM7Pn/DWOrRnxoh+w
        q5F9ZxdZ5D5T1v6kljVwxyfFgHItyyyIL0YS7e2h7cRRscCM+75kMedAP7icb4YN
        LfWBqfKHbHIOqvvQK8T6+Emu/QlG2B5LvuErrop9K0KinhITekpVIO4HCN61cuOe
        CADBKF/5uUJHwS9pWw3uUbpGUwsLBuhJzCY/OpJlDqC8Y9aToi2Ivl5u3/Q/sKjr
        6AZb9lx4q3J2z7tJDrm5MHYwV74elGSXoeoG8nODUqjgklIWAPrt6lQ3WJpO2kug
        8RhCdSbWkcXHfX95AgMBAAGjQjBAMA8GA1UdEwEB/wQFMAMBAf8wHQYDVR0OBBYE
        FOIxhqTPkKVqKBZvMWtKewKWDvDBMA4GA1UdDwEB/wQEAwIBhjANBgkqhkiG9w0B
        AQsFAAOCAQEAqoItII89lOl4TKvg0I1EinxafZLXIheLcdGCxpjRxlZ9QMQUN3yb
        y/8uFKBL0otbQgJEoGhxm4h0tp54g28M6TN1U0332dwkjYxUNwvzrMaV5Na55I2Z
        1hq4GB3NMXW+PvdtsgVOZbEN+zOyOZ5MvJHEQVkT3YRnf6avsdntltcRzHJ16pJc
        Y8rR7yWwPXh1lPaPkxddrCtwayyGxNbNmRybjR48uHRhwu7v2WuAMdChL8H8bp89
        TQLMrMHgSbZfee9hKhO4Zebelf1/cslRSrhkG0ESq6G5MUINj6lMg2g6F0F7Xz2v
        ncD/vuRN5P+vT8th/oZ0Q2Gc68Pun0cn/g==
        -----END CERTIFICATE-----
        -----BEGIN CERTIFICATE-----
        MIIGBTCCA+2gAwIBAgIRAJfKe4Zh4aWNt3bv6ZjQwogwDQYJKoZIhvcNAQEMBQAw
        gZoxCzAJBgNVBAYTAlVTMSIwIAYDVQQKDBlBbWF6b24gV2ViIFNlcnZpY2VzLCBJ
        bmMuMRMwEQYDVQQLDApBbWF6b24gUkRTMQswCQYDVQQIDAJXQTEzMDEGA1UEAwwq
        QW1hem9uIFJEUyBjYS1jZW50cmFsLTEgUm9vdCBDQSBSU0E0MDk2IEcxMRAwDgYD
        VQQHDAdTZWF0dGxlMCAXDTIxMDUyMTIyMDg1M1oYDzIxMjEwNTIxMjMwODUzWjCB
        mjELMAkGA1UEBhMCVVMxIjAgBgNVBAoMGUFtYXpvbiBXZWIgU2VydmljZXMsIElu
        Yy4xEzARBgNVBAsMCkFtYXpvbiBSRFMxCzAJBgNVBAgMAldBMTMwMQYDVQQDDCpB
        bWF6b24gUkRTIGNhLWNlbnRyYWwtMSBSb290IENBIFJTQTQwOTYgRzExEDAOBgNV
        BAcMB1NlYXR0bGUwggIiMA0GCSqGSIb3DQEBAQUAA4ICDwAwggIKAoICAQCpgUH6
        Crzd8cOw9prAh2rkQqAOx2vtuI7xX4tmBG4I/um28eBjyVmgwQ1fpq0Zg2nCKS54
        Nn0pCmT7f3h6Bvopxn0J45AzXEtajFqXf92NQ3iPth95GVfAJSD7gk2LWMhpmID9
        JGQyoGuDPg+hYyr292X6d0madzEktVVGO4mKTF989qEg+tY8+oN0U2fRTrqa2tZp
        iYsmg350ynNopvntsJAfpCO/srwpsqHHLNFZ9jvhTU8uW90wgaKO9i31j/mHggCE
        +CAOaJCM3g+L8DPl/2QKsb6UkBgaaIwKyRgKSj1IlgrK+OdCBCOgM9jjId4Tqo2j
        ZIrrPBGl6fbn1+etZX+2/tf6tegz+yV0HHQRAcKCpaH8AXF44bny9andslBoNjGx
        H6R/3ib4FhPrnBMElzZ5i4+eM/cuPC2huZMBXb/jKgRC/QN1Wm3/nah5FWq+yn+N
        tiAF10Ga0BYzVhHDEwZzN7gn38bcY5yi/CjDUNpY0OzEe2+dpaBKPlXTaFfn9Nba
        CBmXPRF0lLGGtPeTAgjcju+NEcVa82Ht1pqxyu2sDtbu3J5bxp4RKtj+ShwN8nut
        Tkf5Ea9rSmHEY13fzgibZlQhXaiFSKA2ASUwgJP19Putm0XKlBCNSGCoECemewxL
        +7Y8FszS4Uu4eaIwvXVqUEE2yf+4ex0hqQ1acQIDAQABo0IwQDAPBgNVHRMBAf8E
        BTADAQH/MB0GA1UdDgQWBBSeUnXIRxNbYsZLtKomIz4Y1nOZEzAOBgNVHQ8BAf8E
        BAMCAYYwDQYJKoZIhvcNAQEMBQADggIBAIpRvxVS0dzoosBh/qw65ghPUGSbP2D4
        dm6oYCv5g/zJr4fR7NzEbHOXX5aOQnHbQL4M/7veuOCLNPOW1uXwywMg6gY+dbKe
        YtPVA1as8G9sUyadeXyGh2uXGsziMFXyaESwiAXZyiYyKChS3+g26/7jwECFo5vC
        XGhWpIO7Hp35Yglp8AnwnEAo/PnuXgyt2nvyTSrxlEYa0jus6GZEZd77pa82U1JH
        qFhIgmKPWWdvELA3+ra1nKnvpWM/xX0pnMznMej5B3RT3Y+k61+kWghJE81Ix78T
        +tG4jSotgbaL53BhtQWBD1yzbbilqsGE1/DXPXzHVf9yD73fwh2tGWSaVInKYinr
        a4tcrB3KDN/PFq0/w5/21lpZjVFyu/eiPj6DmWDuHW73XnRwZpHo/2OFkei5R7cT
        rn/YdDD6c1dYtSw5YNnS6hdCQ3sOiB/xbPRN9VWJa6se79uZ9NLz6RMOr73DNnb2
        bhIR9Gf7XAA5lYKqQk+A+stoKbIT0F65RnkxrXi/6vSiXfCh/bV6B41cf7MY/6YW
        ehserSdjhQamv35rTFdM+foJwUKz1QN9n9KZhPxeRmwqPitAV79PloksOnX25ElN
        SlyxdndIoA1wia1HRd26EFm2pqfZ2vtD2EjU3wD42CXX4H8fKVDna30nNFSYF0yn
        jGKc3k6UNxpg
        -----END CERTIFICATE-----
    configDb: admin
```

```
    nodes:
      - host: docdb-vbpoyr.ctnrnscupeqf.ca-central-1.docdb.amazonaws.com
        port: 27017
    retryWrites: false
    secretName: documentdb-admin
  reloadCrd:
    fileUpload: -336090572
  settings:
    auth:
      enforce: true
    compliance:
      enforce: true
    registration:
      open: true
    registry: cp.icr.io/cp
```

e) Save the `LicenseService` YAML file.

f) Verify whether the same status conditions are shown for DocumentDB.

6. In the api-licensing pod that is located in **ibm-sls project** > **namespace**, verify that the new connection with DocumentDB is established.
7. Confirm whether the reconciliation is successful.

   If a message `Licensing system has not been initialized, upload a valid entitlement file to enable the Token pool and License Mgmt APIs` is shown, upload the entitlement file again.

   a. Confirm that the api-licensing pod is running with a ready status.

   b. To initialize the licensing system, download the entitlement file. On the Red Hat OpenShift console, from the side navigation menu, select **Workloads** > **Secrets**, select IBM-ls-project, search for `ibm-sls-masocp-<<unique string>>-entitlement`, save, and download the entitlement file.

   c. In the Maximo Application Suite **Setup** page, in the Settings section, the license key checkmark is disabled.

   d. Click **Replace** license file and upload the file that was downloaded in the previous step.

      After the upload is successful, the license key checkmark is enabled.

   e. Confirm the `LicenseService` details are reinitialized.

   f. After the MongoCfg and `LicenseServiceCfg` are configured successfully, verify that the following two collections in the new DocumentDB instance are created.

      - `ibm-sls-<<unique_cluster_string>>_masocp-<<unique_cluster_string>>_licensing` specific to Maximo Application Suite core.

        For example, `ibm-sls-vbpoyr_masocp-vbpoyr_licensing`.

      - `mas_<<unique_cluster_string>>_core` specific to Suite License Service (SLS).

        For example, `mas_vbpoyr_core`.

      The SLS-specific token pools and products collection are also created in the new DocumentDB instance that confirms that the DocumentDB was successfully connected to SLS.

   g. Create an admin user and confirm that the user is shown in the Maximo Application Suite core and Suite License Service collections.

      Note: On the Maximo Application Suite **Configurations** page, in Storage section, select MongoDB to view the configured database details. You cannot update the MongoDB settings on the details page.

      To create an administrator user, add details, such as the display name, user ID, and password. Select **Authorized** for the access type and add entitlements.

      The new user is visible in the Users section of `mas_<<cluster_unique_string>>_core` DocumentDB, which confirms that Maximo Application Suite core is connected with DocumentDB.

      The new user, who is an admin user, is added to the `mas_<<cluster_unique_string>>_core` DocumentDB collection.

8. Delete the existing MongoDB instance by using CLI.

The following example is a sample script to delete the MongoDB instance.

```
# Login to cluster
oc login --token=sha256~xxx --server=https://api.masocp-xupgew.domain.com:6443
# Switch to mongo namespace
oc project ${MONGO_NAMESPACE}

# Note down Certificate resource name
oc get Certificate -n ${MONGO_NAMESPACE}
# Note down StatefulSet name
oc get StatefulSet -n ${MONGO_NAMESPACE}
# Note down mongodbcommunity crd detail
oc get  CustomResourceDefinition mongodbcommunity.mongodbcommunity.mongodb.com -n $
{MONGO_NAMESPACE}
# Note down pvc name
oc get pvc -n ${MONGO_NAMESPACE} | grep mongo

# Delete Certificate
oc delete Certificate mongo-ca-crt -n ${MONGO_NAMESPACE}
oc delete Certificate mongo-server -n ${MONGO_NAMESPACE}

# Delete MongoDBCommunity
oc delete MongoDBCommunity mas-mongo-ce -n ${MONGO_NAMESPACE}
# Delete MongoDB operator
oc delete deployment mongodb-kubernetes-operator  -n ${MONGO_NAMESPACE}
# Delete MongoDB StatefulSet
oc delete StatefulSet mas-mongo-ce
# Delete MongoDB StatefulSet
oc delete CustomResourceDefinition mongodbcommunity.mongodbcommunity.mongodb.com -n $
{MONGO_NAMESPACE}

# Delete Mongo secrets
oc delete secrets --all -n ${MONGO_NAMESPACE}
# Delete Mongo configmaps
oc delete configmaps --all -n ${MONGO_NAMESPACE}
# Delete Mongo pvc
oc delete pvc  data-volume-mas-mongo-ce-0 -n ${MONGO_NAMESPACE}
oc delete pvc  data-volume-mas-mongo-ce-1 -n ${MONGO_NAMESPACE}
oc delete pvc  data-volume-mas-mongo-ce-2 -n ${MONGO_NAMESPACE}
oc delete pvc  logs-volume-mas-mongo-ce-0 -n ${MONGO_NAMESPACE}
oc delete pvc  logs-volume-mas-mongo-ce-1 -n ${MONGO_NAMESPACE}
oc delete pvc  logs-volume-mas-mongo-ce-2 -n ${MONGO_NAMESPACE}
# Delete Mongo project
oc delete project ${MONGO_NAMESPACE}
```

Extracting `Root CA RSA2048` PEM files

If you see a connection error, you can run the following commands to manually add new CA certificates to your trust stores, and update your existing Amazon DocumentDB instances to use the new CA certificates.

1. Run the **`wget`** command to download the `.pem` file for your region.

   For example run the command to download the `.pem` file for Canada (Central) region.

   ```
   wget https://truststore.pki.rds.amazonaws.com/ca-central-1/ca-central-1-bundle.pem
   ```

2. Split the certificate from the certificate names into separate files.

   For example run the following command.

   ```
   cat ca-central-1-bundle.pem lawk 'split_after==1{n++;split_after=0}
   /-----END CERTIFICATE-----/
   {split_after=1} {print > "temp-ca-central-1" n "pem"}'
   ```

3. Search the files that contain the term `Root CA RSA2048`.

   ```
   for cert in $(Is temp-*.pem); do echo $cert; cat $cert | openss| x509 -noout -text |
   grep "Root CA RSA2048 G1"; done
   ```

   Files that contain the term are listed in the output.

4. Copy the file that is required to `root-ca-rsa2048` for your region.

```
cp temp-ca-central-13.pem .../root-ca-rsa2048-ca-central-1.pem
```

5. Verify that the correct `.pem` files are copied.

```
for cert in $(Is root-ca-rsa2048-*pem); do echo $cert; cat $cert | openssl x509 -noout
-text | grep "Root CA RSA2048 G1"; done
```

6. Replace the **`spec.certificates.crt`** and **`spec.mongo.certificates.crt`** parameters with the certificate content from the new PEM file.

Configuring IBM Maximo Application Suite to use Amazon MSK

Configure Amazon Managed Streaming for Apache Kafka (Amazon MSK) so that IBM Maximo Application Suite can process streaming data for applications.

Provisioning Amazon MSK

You can provision Amazon Managed Streaming for Apache Kafka (Amazon MSK) so that IBM Maximo Application Suite processes the streaming data for applications.

Procedure

1. On the Amazon Web Services console, search for `msk`.

   The Amazon MSK service is listed.
2. To open the Amazon MSK console, select **MSK**.
3. In the **Get started** section, click **Create cluster**.

   The **Cluster settings** page is opened.
4. Configure cluster settings, such as creation method, cluster name, cluster type, Apache Kafka version, brokers, and cluster configurations.
   a) To customize settings, select **Custom create as the Creation method**.
   b) Assign a name for your cluster, for example, `aws_msk`.
   c) To specify the number of brokers and the amount of storage that is required for each broker, select **Provisioned as the Cluster type**.
   d) Select the recommended 2.8.1 Apache Kafka version.
   e) In the Brokers section, select three zones and one broker for each zone so that your cluster has three total brokers that are distributed evenly across your three availability zones..
   f) In the Configurations section, select **Custom configuration**.
   g) On the **Create configuration** page, enter the configuration name and specify the configuration properties.

      You must append the `allow.everyone.if.no.acl.found` property and set it to `true` in the Apache Kafka code.
   h) Click **create**.

      The cluster configuration is created and listed on the Amazon MSK console.
   i) To finish the configuration, on the MSK console, click your new cluster configuration.

      You return to the **Configuration** page.
   j) Click **Next** to configure the network settings.
5. To deploy your brokers, configure network settings, such as Virtual Private Network (VPC), zones, subnet, and security groups.
   a) Select the virtual networking environment for your cluster.
   b) Select the subnet for each zone that was configured.

c) To create the security groups, which act as a virtual firewall for your instances to control inbound and outbound traffic, click **create**.

Before you click create, ensure that you remove the default security group from the list of Chosen security groups.

d) On the **Create security group** page, add the security group name and description and select the VPC unique string that was created earlier.

e) For inbound and outbound rules, select **Custom TCP** for the rule type and **Custom** for the source.

The security group is created.

f) Return to the **Create security group** page to select your custom security group, and click Choose.

g) Click **next** to add the security settings.

h) In the Security settings, select **SASL/SCRAM authentication** as your access control method and create a custom managed key to encrypt your data.

i) Select details, such as Symmetric for the key type, encrypt and decrypt as key usage, KMS as key material origin, and single-region key as region.

j) Customize your key configurations for alias, tags, administrators, and usage permissions.

k) Assign the key policy and click **finish** to complete the custom key configuration.

6. On the **Monitoring and tags** page, select basic monitoring and click **next**.
7. On the **Review and create** page, view and verify the configured parameters and click **Create cluster**.

The custom cluster is provisioned and shown on the MSK console.

8. To associate the Amazon Web Services Secrets Manager secrets with the cluster to configure SASL/SCRAM authentication, click **Associate secrets**.
   a) On the **Associate secrets** page, click **Create secret**.
   b) On the **Choose secret type** page, select the encryption secret key and provide details, such as secret type and key or value pairs.
   c) On the **Configure secret** page, enter the secret name.
   d) Optional: Configure rotation details for the secret, such as automatic rotation, rotation schedule, and rotation function.
   e) Review the secret information, such as secret type and sample code.
   f) Click **store**.
   g) On the **Associate secrets to cluster** page, choose the secret that was created and click **Associate secrets**.

   For more information, see https://docs.aws.amazon.com/msk/latest/developerguide/msk-password.html
9. On the MSK console, verify that the client information is added.

What to do next

Configure the IBM Maximo Application Suite to use a provisioned Amazon MSK instance.

For more information, see "Configuring IBM Maximo Application Suite with Amazon Web Services MSK" on page 70.

Configuring IBM Maximo Application Suite with Amazon Web Services MSK

You can configure Maximo Application Suite to use a provisioned Amazon MSK instance.

Procedure

1. Log in to Maximo Application Suite as an admin user.
2. On the **Configurations** page, select the Apache Kafka option from others list.
3. To add Apache Kafka under the system scope, click **Configure**.

4. Add parameters, such as the hosts, port, SASL mechanism as scram-aha-512, username, and password.
5. Paste the Amazon ca-certificate details that you can copy from https://www.amazontrust.com/repository/AmazonRootCA1.pem.
6. Click **Confirm** and then **Save**.
7. Verify the Kafka configuration status in **Configurations** > **Apache Kafka** > **System scope**.
8. Optional: To verify that the Kafka configuration is running, open the instances tab of the `KafkaCfg` CRD for the selected namespace on the Red Hat OpenShift console.

Deploying and activating IoT and Maximo Monitor

After you provision Amazon MSK and configure it to work with IBM Maximo Application Suite, you can deploy and activate Apache Kafka from the IoT tool and Maximo Monitor.

Procedure

1. Log in to Maximo Application Suite as an admin user.
2. On the **Catalog** page, on the **Tools** tab, select IoT.
3. On the **IoT** page, from the toolbar, click the **Administration** icon.
4. Click **Continue** to open the **Administer application upgrades** page.
5. Select an upgrade strategy and subscribe to a channel.
   For example, select **Channel subscription** and subscribe to the 8.x channel.

   The IoT tool deployment begins.
6. After the IoT tool is deployed successfully, on the **Deploy IoT** page, configure the database connection.
7. Click **Exit**.
8. On the **Configurations** page, to view database details, from the **Storage** list, select the database connection.

   You must configure the database connection by adding the Java database connectivity values at the system scope level. The information for connectivity is available in the IBM Maximo Manage Workspace-application scope.
9. Select the Workspace-application scope to edit the JDBC connection information, such as the connection string, username and password, and certificate content.

   Tip: You can find the username and password from the JDBC-DB2® namespace on the **Secrets** page on the Red Hat OpenShift console.
10. Click **Save**.
11. Verify that the connection is configured on the Red Hat OpenShift console **CustomResourceDefinitions** page and the Maximo Application Suite **Configurations** > **Database connection** page.
12. Deploy and activate the IoT tool.

    For more information, see Deploying the IoT tool.
13. Repeat the steps for Maximo Monitor.

    For more information, see Deploying IBM Maximo Monitor and Activating IBM Maximo Monitor.

What to do next

Verify that the Apache Kafka connection is established in the Maximo Monitor application by using the IoT device simulator.

For more information, see “Verifying Apache Kafka connection with IoT simulator” on page 72.

Verifying Apache Kafka connection with IoT simulator

Verify that the Apache Kafka connection is established from the Maximo Monitor application by using the IoT device simulator.

Procedure

1. Log in to Maximo Application Suite as an admin user.
2. Create a user by providing details, such as identity, entitlements, and application access to Maximo Monitor and the IoT tool.
3. On the **Catalog** page, on the toolbar, click the **AppSwitcher** icon.
4. Select **Maximo Monitor** and create device types and add metrics.
5. Return to the **Catalog** page, and on the toolbar, click the **AppSwitcher** icon.
6. Select **IoT** and create a device simulator.
7. Configure the device simulator by selecting a device type, configuring the event and payload, and adding device.
8. To verify that the simulation was successful, view the number of events that are shared in Simulation.

IBM Maximo Application Suite installation with Microsoft Azure Resource Manager templates

You can install Maximo Application Suite in the Microsoft Azure cloud by using the Microsoft Azure Resource Manager templates. In Microsoft Azure Marketplace, you subscribe to Maximo Application Suite, configure the installation parameters, and install the application. The network infrastructure, Red Hat OpenShift cluster, and Maximo Application Suite components are created in your Microsoft Azure cloud account.

When you select the **New OpenShift cluster (IPI)** option, the network infrastructure, Red Hat OpenShift cluster, and Maximo Application Suite components are created in your Microsoft Azure cloud account. You can reuse existing Red Hat OpenShift cluster cluster from IPI or UPI with the **Existing OpenShift cluster** option.

Maximo Application Suite is available as a bring-your-own-license (BYOL) and contract pricing product in Microsoft Azure Marketplace. After you configure the installation requirements and consider your installation preferences, you subscribe to the product, specify the installation parameters, and start the installation.

In your Microsoft Azure account, the installation process creates the virtual network infrastructure, the Red Hat OpenShift cluster, the application prerequisites, and the application itself.

If you provide the SMTP configuration during the deployment, you receive emails that contain the information that you need to access Maximo Application Suite.

Installing IBM Maximo Application Suite on Microsoft Azure

The IBM Maximo Application Suite installation process involves configuring the installation prerequisites, finalizing your installation preferences, installing the application, and completing the initial application setup.

Before you begin

You must configure prerequisites and gather the information that you need to specify the installation parameters. For more information, see Planning to install on Microsoft Azure.

In addition, you must consider your installation preferences, such as the type of offering that you want and whether you want to create an Red Hat OpenShift cluster or reuse an existing one.

Note: The existing cluster must have been created by using the automated deployment option.

For more information, see Installation considerations.

Installing Maximo Application Suite

To install the Maximo Application Suite on Microsoft Azure, you configure the prerequisite components, consider your installation preferences, specify the installation criteria in parameters that are provided during the deployment, and deploy the product.

Before you begin

Before you can install Maximo Application Suite on Microsoft Azure, you must configure prerequisites and gather information that you need to complete the installation. For more information, see Prerequisites for installing Maximo Application Suite on Microsoft Azure.

You must also consider other criteria, such as the type of Maximo Application Suite offering that you want and whether you want to create an Red Hat OpenShift cluster or reuse an existing one.

Note: The existing cluster must be created by using the automated deployment option only.

If you provide the existing Red Hat OpenShift cluster, the installation process checks if any of the following products are already installed:

- Red Hat OpenShift 4.10.35
- IBM Cloud Pak foundational services 4.6.0
- IBM Cloud Certificate Manager 3.21.1 or higher
- MongoDB (CE) 4.2.6 or higher
- IBM Suite License Service 3.4.0 or higher
- IBM Data Reporter Operator
- IBM Cloud Pak for Data 4.0.9
- IBM Maximo Application Suite 8.10
- IBM Maximo Manage 8.6

For more information, see Preparing to installing Maximo Application Suite on Microsoft Azure.

Installing BYOL IBM Maximo Application Suite

You can install Maximo Application Suite in Microsoft Azure.

About this task

To install Maximo Application Suite in Microsoft Azure, the following three fulfillment options are available.

1. Existing Red Hat OpenShift cluster
2. New Red Hat OpenShift cluster by using the Installer Provisioned Infrastructure (IPI)
3. New Red Hat OpenShift cluster by using the User Provisioned Infrastructure (UPI)

After you select the fulfillment options from Microsoft Azure Marketplace, complete the following steps.

Procedure

1. In the Microsoft Azure Marketplace, use the search function and search for `Maximo Application Suite (BYOL)`
2. Open the Suite product.
3. Review the product information and click **Subscribe**.
4. In the Subscribe to IBM Maximo Application Suite (BYOL) step, enter the installation parameters by using the information that you gathered when you configured the Prerequisites for installing Maximo Application Suite on Microsoft Azure and considered your Preparing to installing Maximo Application Suite on Microsoft Azure.

Basics (Required)

- Resource group

  This resource group is a new boot node resource group, as the boot node is created in this resource group. The Red Hat OpenShift cluster is created in its own resource group that is created by the Red Hat OpenShift installer when you use the New OpenShift cluster (IPI) option.

  From Maximo Application Suite 8.8 or later, when you use the 'New OpenShift cluster, existing network (UPI) option, the Red Hat OpenShift cluster resources are created in the resource group where the existing VNet is configured.
- Region
- Subscription Id
- Public key value
- Boot node NSG Ingress CIDR range
- Bootnode Subnet CIDR IP range

  The `Bootnode Subnet` Classless Inter-Domain Routing (CIDR) IP range is required for UPI and existing OCP deployments.

  Tip: The `bootnode subnet` CIDR range should be in the `Vnet` CIDR range.

  Starting in 8.10, if you are reusing existing Red Hat OpenShift cluster or starting in 8.11, if you are using existing Microsoft Azure Red Hat OpenShift cluster, the following options are available.

  – BootNodeVnetId

  – BootNodeVnetResourceGroup
- Microsoft Azure service principal ID
- Microsoft Azure service principal client secret

Application Settings (Optional)

- Public or hosted domain. This is required only if you are provisioning new Red Hat OpenShift cluster in IPI or UPI mode. For an existing Red Hat OpenShift cluster, public domain field is not available.

  Public domain is changed to **Hosted domain** from Maximo Application Suite 8.10 or later.

  If you provision a new Red Hat OpenShift cluster by using the IPI option, the hosted domain value is fetched from the Public DNS zone. If you provision an existing Red Hat OpenShift cluster by using the UPI option, the hosted domain value is fetched from the Private DNS zone for installing a private cluster and DNS Zone for a public cluster.
- Offering type
- From Maximo Application Suite 8.8 or later, Cluster size. This is required only if you are provisioning new Red Hat OpenShift cluster in IPI or UPI mode.
- Entitled registry key
- Red Hat OpenShift pull secret . This is required only if you are provisioning new Red Hat OpenShift cluster in IPI or UPI mode. If using existing Red Hat OpenShift cluster, keep the value empty.
- Maximo Application Suite license URL
- From Maximo Application Suite 8.9 or later, Operational Mode - specify the parameter as Production or Non-production.

  You can use the non-production installations for internal development and testing. The installation AppPoints are unused in the non-production installations. These specifications are also visible in the metrics shared with IBM as well as on the product UI.

Existing network infrastructure

Existing VNet to use, For more information, see SSH key pair.

Existing Infrastructure

The following information is required only if you are reusing existing Red Hat OpenShift cluster.

- Red Hat OpenShift cluster API URL.
- Red Hat OpenShift user
- Red Hat OpenShift password

For more information, see Existing network infrastructure by using Red Hat OpenShift UPI mode.

- SLS endpoint URL
- SLS registration key
- SLS public certificate URL
- DRO endpoint URL
- DRO API key
- DRO public certificate URL

Database Settings

- Maximo Application Suite Manage DB user
- Maximo Application Suite Manage DB password
- Maximo Application Suite Manage DB JDBC URL
- Maximo Application Suite Manage DB certificate URL
- Starting in 8.11, VNetId of the database that is provisioned
- Import demo data

Note: If you choose to install Maximo Application Suite with Manage, you can use the default IBM Db2 instance that is provisioned by IBM instead of configuring your own external Db2 instance. To configure the default IBM Db2, do not add information in the username, password, JDBC URL, certificate URL, and demo data fields.

The internal Db2 configuration is available from Maximo Application Suite 8.10 or later.

Email Settings

- Email notification
- SMTP host
- SMTP port
- SMTP username
- SMTP password
- Notification email addresses

5. To begin the installation, click **Review +** > **Create**.

Installing customer managed IBM Maximo Application Suite

You must first subscribe to IBM Maximo Application Suite to complete the transactional aspect of the IBM Maximo Application Suite customer managed purchase process.

About this task

You can install either by using the public paid offer or private paid offer.

Procedure

- **Installing with public paid offer**
  a) In the Microsoft Azure, use the search function and search for Maximo Application Suite (customer-managed).

b) Open the IBM Maximo Application Suite product.

c) Review the product information and click **Subscribe**.

d) In the Subscribe to IBM Maximo Application Suite (customer-managed) step, enter the parameters based on the following guidelines.

Basics (Required)

- Resource group: This resource group is used to keep the application instance representing the purchased instance of IBM Maximo Application Suite (customer-managed) product.
- Region
- Microsoft Azure location
- Name: Name of the application instance representing the purchased IBM Maximo Application Suite (customer-managed) product.
- Recurring billing: Whether you want the renew or terminate the billing after subscription is ended.

e) To review and initiate the purchase process, click Review and Subscribe.

Note: The public offer is a fixed contract of 12 months with 500 AppPoints.

f) To initiate the purchase process, click Subscribe.

g) You see the message that subscription is in progress.

h) After the subscription process completes, you will get an auto-generated email from Microsoft to activate IBM Maximo Application Suite (customer-managed) subscription.

i) You can click the **Activate now** in the email or **Configure account now** available for the offer. Clicking on the button will take you to the IBM registration page where you will need to provide the following details.

Company
Your company name.

Full name
Your full name.

Email address
Your corporate email address.

IBM ID
Email address that you have registered as your IBM ID.

Offer type
Select the public offer type.

IBM Quote number
Leave this empty. This is required only for private offer.

Company address
This is required for the public offer.

State
State where the company is located. This is required for the public offer.

Zip code
Zip code of the address. This is required for the public offer.

Country
Country where the company is located. This is required for the public offer.

j) Click the COMPLETE REGISTRATION button.

You will get an auto-generated email from Tackle.io about the process initiation for setting up your account.

k) After the account setup is completed, you will get two emails from IBM. First email describes how to retrieve IBM Entitled Registry key from My IBM. Second email describes how to get the IBM Maximo Application Suite license and Red Hat OpenShift pull secret.

l) After these artifacts are retrieved, perform the actual product deployment by following the installations steps for IBM Maximo Application Suite (BYOL) product.

- **Installing with private paid offer**

a) Contact your IBM Sales Representative for a customized subscription contract (Private Offer) with recommended configuration and negotiated pricing.

For more information, see https://www.ibm.com/products/maximo/pricing.

b) After you agree upon the price and number of AppPoints, IBM sends a link so that you can accept the private offer.

c) Only users with the permissions shown in this table can access and sign the offer contract.

- For Microsoft Customer Agreement accounts, the user role must be billing account owner or billing account contributor.
- For EA accounts, the user role must be EA admins.

For more information about the type of account you have, see Billing accounts and scopes.

Accepting a private offer simply means you've agreed to the terms and prices listed in the offer. No purchase has been made and no money has exchanged yet.

d) After the private offer is accepted, it will appear in the marketplace under **My Marketplace** > **Private products** section.

e) Select the appropriate offer in case you have multiple private offers for same or different products.

f) Review the product information and click Subscribe.

g) In the Subscribe to IBM Maximo Application Suite (customer-managed) step, enter the parameters based on the following guidelines.

Basics (Required)

- Resource group: This resource group is used to keep the application instance representing the purchased instance of IBM Maximo Application Suite (customer-managed) product.
- Region
- Microsoft Azure location
- Name: Name of the application instance representing the purchased IBM Maximo Application Suite (customer-managed) product.
- Recurring billing: Whether you want the renew or terminate the billing after subscription is ended.

h) To review and initiate the purchase process, click Review and Subscribe.

i) To initiate the purchase process, click Subscribe.

j) You see the message that subscription is in progress.

k) Once subscription process completes, you will get an auto-generated email from Microsoft to activate IBM Maximo Application Suite (customer-managed) subscription.

l) You can click **Activate now** in the email or **Configure account now** available for the offer. Clicking on the button will take you to the IBM registration page where you will need to provide the following details.

Company
Your company name.

Full name
Your full name.

Email address
Your corporate email address.

IBM ID
Email address that you have registered as your IBM ID.

Offer type
Select the public offer type.

IBM Quote number
Provide the IBM quote number

Company address
Leave this empty. This is required only for the public offer.

State
Leave this empty. This is required only for the public offer.

Zip code
Leave this empty. This is required only for the public offer.

Country
Leave this empty. This is required only for the public offer.

m) Click the COMPLETE REGISTRATION button.

You will get an auto-generated email from Tackle.io about the process initiation for setting up your account.

n) After the account setup is completed, you will get two emails from IBM. First email describes how to retrieve IBM Entitled Registry key from My IBM. Second email describes how to get the IBM Maximo Application Suite license and Red Hat OpenShift pull secret.

o) After these artifacts are retrieved, perform the actual product deployment by following the installations steps for IBM Maximo Application Suite (BYOL) product.

Installing Cloud Pak for Data on an Azure instance of Maximo Application Suite

To install Cloud Pak for Data on an existing Maximo Application Suite instance on Azure, you clone a Git repository and run a script.

About this task

If you installed Maximo Application Suite core and Maximo Manage, your Maximo Application Suite instance does not include Cloud Pak for Data. If you want to deploy certain applications and add-ons in the Suite, such as Maximo Monitor, you must first install Cloud Pak for Data.

To install this application, you clone a Git repository, locate the installation script, and run it. When you run the script, you must provide the Azure deployment values that identify the Maximo Application Suite instance, such as boot node's resource group, and your entitled registry key. The installation script retrieves the Suite's `<unique-string>` and the Red Hat OpenShift cluster details from the resource group's deployment object. If the cluster credentials, that is the username and password, are changed since you installed the Suite, you must provide the updated credentials when you run the script.

You run the script on your local machine or on the boot node in the Suite's Red Hat OpenShift cluster.

Installing Cloud Pak for Data

Complete the following steps to install Cloud Pak for Data in your Maximo Application Suite instance:

Procedure

1. On the machine where you want to run the script, in a command shell, log in to the Azure service by running the following command.

   ```
   az login
   ```

   It opens a new browser window where you can log in with the Azure credentials. After login, you can close the browser window and continue the next steps from the command shell.
2. Clone the Git repository that contains the script by running the following command:

```
git clone https://github.com/ibm-mas/multicloud-bootstrap.git
```

3. Make the script executable by running the following commands:

```
cd multicloud-bootstrap/azure
chmod +x deploy-cp4d.sh
```

4. View the script's usage information by running the following command:

```
./deploy-cp4d.sh -h
```

5. Specify the required options and run the script.
   - Use the `r` option to specify the resource group name of the boot node, for example: `-r mas-ocp-deploy-rg`
   - Use the e option to specify the entitled registry key that you provided when you installed the Suite, for example: `-e <entitlement-key>`
   - Use the u and p options to specify the Suite's Red Hat OpenShift cluster credentials, for example: `-u <ocp-user> -p <ocp-password>`
   - The following sample command installs Cloud Pak for Data by using all of these example options:
   - The script takes 60 minutes to

     ```
     ./deploy-cp4d.sh -r <bootnode-resource-group> -e <entitlement-key>-u <ocp-user> -p <ocp-password>
     ```

     installation Cloud Pak for Data into the Maximo Application Suite instance.
   - Verify that the script completed successfully. If the script is successful, output that is similar to the following text is displayed:

     ```
     :: Script Inputs ::
       Resource group = mas-ocp-deploy-rg
     …
     :: OpenShift Details ::
     …
     OpenShift Login is successful.
     …
     ==== MAS configuration started ====
     ==== MAS configuration completed ====
     ==== Execution completed at Mon Mar 25 14:02:56 IST 2022 ====
     ```

   - Maximo Application Suite core with Cloud Pak for Data is deployed.
   - Db2 Warehouse service of Cloud Pak for Data is enabled.

     For information about troubleshooting, see IBM Support notes for zen-databases failure.
6. Login to the Cloud Pak for Data admin console.
7. Select the service catalog and search for Db2 Warehouse.
8. Click **Provision instance** of Db2 Warehouse service.
9. Configure the database.

What to do next

You can now deploy the following applications and industry solutions that depend on Cloud Pak for Data:

- Maximo Application Suite
- Maximo Monitor /IoT tool
- Maximo Health
- Maximo Predict
- Maximo Collaborate
- Maximo Health and Predict - Utilities

Monitoring installation logs

During the installation process, the ARM template that you configured is used to create a boot node. The boot node contains all of the required resources to complete the installation. To verify that the boot node is created successfully, in the **Deployment** section of the boot node resource group, you will see a link similar to *3 Succeeded*. After clicking that link, you will see a deployment that is related to the operation you submitted from the Marketplace. Click that deployment and make sure it is completed successfully. This is only the boot node creation part. The actual Maximo Application Suite deployment continues running in the background on the boot node.

Procedure

- **Monitoring the installation logs using Azure Log Analytics**
  a) In the Azure **Log Analytics workspaces** service, Click the workspace created in the boot node resource group.
  b) Click **Logs** from the side navigation in the workspace window.
  c) Close the two pop-ups that appear by default. First is for **Welcome to Log Analytics**, and second is for **Queries**.
  d) Expand the **New Query 1** > **Custom Logs** section.
  e) Click the log table name (there would be only one), and click **Run**.
  f) The logs from the installation log file are shown in the table format.

  The logs do not update automatically, you will need to refresh the page to get the updated logs.

  a. You can export the logs to the `.csv` file to review in detail.
  b. Once you use the message of either provisioning completed (for successful deployment) or failed (for failed deployment), the deployment can be treated as complete.

```
===== PROVISIONING COMPLETED =====
```

  OR

```
===== PROVISIONING FAILED =====
```

- **Monitoring the installation logs in the boot node**
  a) Connect to the boot node by using Secure Shell (SSH) access. For instructions, see Accessing the boot node and Red Hat OpenShift cluster.
  b) Run the following command to switch to the root user:

```
sudo su -
```

  c) Monitor the installation log updates by running the following command:

```
tail -f /root/ansible-devops/multicloud-bootstrap/root/mas-on-aws/mas-provisioning.log
```

Results

Proactively check the status of the deployment. Depending on the parameters that you specified, the installation time might vary.

If the installation is unsuccessful, use the information in the Troubleshooting installation problems topic to identify and resolve the problem.

Accessing IBM Maximo Application Suite

After you install Maximo Application Suite, you need the following items of information to access it:

- The Maximo Application Suite administrator URL, which you use to connect to Maximo Application Suite through a browser.
- Your username and password.
- The public certificate for Maximo Application Suite. You import this certificate into your browser's trusted store to ensure secure communication between your browser and Maximo Application Suite.

How you retrieve these items of information depends on whether you opted for the email notification during the deployment.

If you have provided the correct SMTP configuration during the deployment, you can retrieve the administrator URL, username, and password from the emails that you received. The public certificate is attached to these emails.

If you have not opted for the email notification, you can retrieve the administrator URL and credentials of the Red Hat OpenShift cluster from the deployment created in the boot node's resource group. However, to retrieve your username, password, and the public certificate of Maximo Application Suite, you must connect to the Red Hat OpenShift cluster.

Procedure

Complete the following steps to retrieve all of the required information:

1. In the **Outputs** section of the deployment that was created in the bootnode resource group, record the following values:
   - The value of the `masAdminUrl` key. This value contains the administrator URL.
   - The values of the `openShiftConsoleUrl`. These values contain the URL for the Red Hat OpenShift console.
   - The value of the `clusterUniqueString` key. This value contains the cluster unique string to look for the correct resources in Red Hat OpenShift.
2. The Microsoft Azure Vault contains the credentials for Red Hat OpenShift cluster. Look for the vault named `maximo-vault-<unique-string>` in the Microsoft Azure Key Vaults service. It consists of two secrets named `maximo-ocp-secret` containing Red Hat OpenShift credentials and `maximo-mas-secret` containing Maximo Application Suite credentials. Retrieve the Red Hat OpenShift Container Platform credentials from this secret.

   Note: By default no user has access to view the secrets. You must provide appropriate access from the **Access policies** section of the key vault.
3. Connect to the Red Hat OpenShift cluster.
4. Select **Workloads** > **Secrets** from the navigation page.
5. Select the `mas-<unique-string>-core` project.
6. Click the `<unique-string>-credentials-superuser` secret.
7. Click the 'Reveal values' link to get the username and password for Maximo Application Suite.
8. Click the `<unique-string>-cert-public` secret from the `mas-<unique-string>-core` project.
9. Click the 'Reveal values' link to get the contents of the certificates.
10. Retrieve the contents of `ca.crt` file, which is the public certificate for Maximo Application Suite.
11. After you import the public certificate into your browser's trusted store, paste the Maximo Application Suite administrator URL into your browser and enter the authentication credentials to access the application.

Results

You can now log in to Maximo Application Suite to deploy applications, create users, and specify configuration.

In addition, if you installed the Maximo Application Suite core, Maximo Application Suite core with Cloud Pak for Data is deployed.

- IBM Db2 Warehouse service of Cloud Pak for Data is enabled.

To configure the Cloud Pak for Data database:

1. Login to the Cloud Pak for Data admin console.
2. Select the service catalog and search for Db2 Warehouse.
3. Click Provision instance of Db2 Warehouse service.
4. Configure the database.

If you installed `Maximo Manage` offering type, and you want to deploy certain applications and add-ons in Maximo Application Suite, such as Maximo Monitor, you must first install Cloud Pak for Data. For more information, see Installing Cloud Pak for Data on an Amazon Web Services instance of Maximo Application Suite.

Configuring Let's Encrypt for Maximo Application Suite on Microsoft Azure

When you install Maximo Application Suite on Microsoft Azure, Maximo Application Suite uses self-signed certificates. If you want to use well-known certificates signed by Certificate Authority as Let's Encrypt, you can install and configure Let's Encrypt on Microsoft Azure.

Before you begin

To configure Let's Encrypt, a service principal in Microsoft Azure must be created. For more information, see https://cert-manager.io/docs/configuration/acme/dns01/azuredns/#service-principal

About this task

The cert-manager can create and then delete **`DNS-01`** records in Microsoft Azure DNS. However, the DNS needs to authenticate with Microsoft Azure first. The following method uses the Microsoft Azure Service principal authentication to configure Let's Encrypt.

Procedure

1. Create a service principal in Microsoft Azure by using the service principal connection parameters. For example:

```
AZURE_DNS_ZONE_RESOURCE_GROUP=masperf

AZURE_DNS_ZONE=mas4azure.com

AZURE_CERT_MANAGER_SP_APP_ID=3xx721x5-xx3x-4x10-x39x-xxx31335405

AZURE_CERT_MANAGER_SP_PASSWORD=X87xXxX6q6xxXxXxx7nYhZmbXxxxX~Tho

AZURE_TENANT_ID=xxx67057-50x9-4xx4-98x3-xxxx64xxx9x9

AZURE_SUBSCRIPTION_ID=x2xx5467-2502-4b05-x78x-744604x6531x
```

   Tip: For all examples, replace the parameters given in the example with your own parameters.

2. Log in to Microsoft Azure by using the service principal connection details.

```
az login --service-principal -u $AZURE_CERT_MANAGER_SP_APP_ID -p
$AZURE_CERT_MANAGER_SP_PASSWORD --tenant $AZURE_TENANT_ID
```

3. Create the **`DNS Contributor`** role to associate DNS zone with the service principal. For example:

```
DNS_ID=$(az network dns zone show --name $AZURE_DNS_ZONE --resource-group
$AZURE_DNS_ZONE_RESOURCE_GROUP --query "id" --output tsv)

az role assignment create --role "DNS Zone Contributor"  --assignee-object-id
$AZURE_CERT_MANAGER_SP_APP_ID --assignee-principal-type ServicePrincipal --scope $DNS_ID
```

4. In **DNS Record**, click your domain, create **A** record set `*.<<Cluster_unique_String>>.<<DNS_NAME>>`.
   For example, `*.i4l7mh.mas4azure.com`, where **i4l7mh** is the cluster unique string and **mas4azure** is the DNS name.

   Tip: Use the same value that is used in the **A** record.
5. Login to the Red Hat OpenShift cluster.
   For example:

```
oc login --token=<<token_number>> --server=https://api.masocp-i4l7mh.mas4azure.com:6443
```

6. Create a secret **azuredns-config**, which contains the service principal password.
   For example:

```
oc create secret generic azuredns-config --from-literal=client-
secret=$AZURE_CERT_MANAGER_SP_PASSWORD -n ibm-common-services
```

7. In the Red Hat OpenShift console, create a **ClusterIssuer** from the **Instances** tab of the **Home** > **API Explorer** `cert-manager.io` group.
   For example:

```
apiversion: cert-manager.io/v1
kind: ClusterIssuer
metadata:
  name: letsencrypt-prod
  namespace: ibm-common-services
spec:
  acme:
    server: https://acme-v02.api.letsencrypt.org/directory
    email: username.ibm.com
    privateKeySecretRef:
      name: letsencrypt-prod
    solvers:
      - dns01:
        azureDNS:
          clientID: 3xx721x5-xx3x-4x10-x39x-xxx31335405
          clientSecretSecretRef:
            name: azuredns-config
            key: client-secret
          subscriptionID: x2xx5467-2502-4b05-x78x-744604x6531x
          tenantID: xxx67057-50x9-4xx4-98x3-xxxx64xxx9x9
          resourceGroupName: masperf
          hostedZoneName: mas4azure.com
          environment: AzurePublicCloud
```

   Note: In Maximo Application Suite 8.10, wait for the routes to regenerate and verify the generated routes to check if the certificate is signed by Let's encrypt.
8. In the Red Hat OpenShift console, from **Home** > **API Explorer** search for a suite in your namespace.
9. In the Red Hat OpenShift console from **Administration** > **CustomResourceDefinition**, select the **Instances** tab for your **Suite CRD**.
10. Click your Custom Resource, and in the **Instances** tab, select **YAML** to add **cluster issue** and **domain** parameters in the **spec** section.

```
---
spec:
  certificateIssuer:
    duration: 8760h0m0s
    name: prod-route53-issuer
    renewBefore: 720h0m0s
  domain: <<masinstance_id>>.<<domain>>
```

11. Delete the **finalizer** section from the same Suite YAML to force a reconciliation, and then save the YAML file.

```
finalizers:
  - core.mas.ibm.com/finalizer
```

12. In **Networking** under **Routes** of project *mas-<mas_instance_id>-core*, wait for the **Routes** to regenerate for the namespace.

    Note: The **Routes** regeneration takes some time.

    The Certificate in the routes is signed by Let's encrypt.
13. Login to the Maximo Application Suite administrator screen and verify the certificate signer.

Administering Maximo Application Suite on Microsoft Azure

Use the **Administration** page to configure and manage the suite level features.

You access the **Administration** page from the **Suite** application in the side navigation menu.

Getting started

The first time that you log in to Maximo Application Suite as the administrator user, no applications are deployed and no users are logged in to the system.

Complete the following tasks to get your Maximo Application Suite environment ready:

- Use the **Catalog** to deploy and activate one or more applications.
- Use the Users page to add and manage Maximo Application Suite administrators and application users.
- Use the **Configurations** page to administer suite-level configurations. For example, if you deployed the Maximo Manage application, you can add or update the database configuration information.
- If you installed Maximo Application Suite core and Cloud Pak for Data its service Db2 Warehouse is enabled by default.
- If you install Maximo Manage, and you want to deploy certain applications and add-ons in the Maximo Application Suite, such as Maximo Monitor, you must first install Cloud Pak for Data.

  For more information, see "Installing Cloud Pak for Data on an Azure instance of Maximo Application Suite" on page 78.

Boot node and cluster administration

For operational reasons, you might need command-line access to the boot node, the bastion host, or the cluster nodes that are located in the Maximo Application Suite Microsoft Azure Virtual Network (VNet).For more information, see Accessing the boot node and Red Hat OpenShift cluster and boot node and bastion node.

Maximo Application Suite authentication

The default authentication uses the Maximo Application Suite authentication. However, you can change the authentication to use LDAP or any other supported authentication method.

For more information, see Authentication methods.

Accessing the boot node and Red Hat OpenShift cluster

By using Secure Shell (SSH) public key authentication, you can access the boot node and the Red Hat OpenShift cluster nodes.

About this task

The Maximo Application Suite deployment configures the Azure Bastion service to provide the SSH access to the Red Hat OpenShift cluster nodes.

1. Go to resource group where the Red Hat OpenShift Container Platform cluster virtual machines are deployed, for example, masocp-<uniqustr>-rg.

   If you use the New OpenShift cluster (IPI) deployment mode, the resource group name format is `masocp-<uniqustr>-rg`. For the New OpenShift cluster, existing network (UPI) deployment mode, the resource group is the same as the one where existing network infrastructure is available, of which the VNet was provided at the time of deployment.

2. Select the virtual machine to which you want to connect and choose to connect option on azure portal.
3. Choose bastion option from the connect menu and it takes you to the page where you need to fill the details like username and SSH private key.
4. Use the **“core”** username. You can either upload your private key from local machine or you can paste the content of private key on azure portal textbox.
5. Press connect button. You are redirected to a new browser window where you can access the VM command line.

In the Azure cloud, when you start the Maximo Application Suite installation, a boot node is created. By using the required tools and the installation parameters, the boot node completes the installation.

In the Red Hat OpenShift cluster that is created during the installation. The boot node has the public IP address so it can be accessed directly from outside using SSH client. However, the Red Hat OpenShift cluster nodes are not assigned with the public IP address.

The boot node, and private cluster nodes are all Azure virtual machines. If you need to perform maintenance or troubleshooting tasks in a virtual machine, you can connect to it by using Azure Bastion service.

Before you installed Maximo Application Suite, you generated a key pair, which consists of a public key and a private key. You stored the private key locally. When you specified the installation parameters, you selected the public key in the `sshKey` parameter.

During the installation, a copy of the public key is stored in the boot node, and the private cluster nodes. Because you have the corresponding private key, you can access these instances by using SSH over Bastion service.

Procedure

Accessing the boot node and private cluster nodes

To use SSH access to connect to the boot node and the private cluster nodes, complete the following steps:

1. In your Azure account, go to the Virtual machines service.
2. Click the virtual machine that belongs to the Red Hat OpenShift cluster you have provisioned.
   - You can find the virtual machine based on the resource group name where the virtual machine is available. The resource group name will start with `masocp-<unique-identifier>`.
   - All the virtual machines are in the same resource group. In case of the IPI deployment mode, the resource group name format is `masocp-<uniqustr>-rg`. In case of UPI deployment mode, the resource group is the same as the one where existing network infrastructure is available, of which the VNet was provided at the time of deployment.
3. Retrieve the node details.

   For bootnode

   a. To retrieve the boot node details, get the public IP address from *Public IP address field* displayed on the virtual machine’s *Overview* page.

   b. Perform the SSH from your workstation.

```
ssh azureuser@<bootnode-public-ip-address>
```

   For Red Hat OpenShift cluster nodes

   a. Choose *Bastion* option from the *Connect* menu and it will take you to the page where you need to fill the details like Username and SSH private key.

   b. Use username as **“core”** and you can either upload your private key from local machine or you can paste the content of private key on azure portal text-box.

   c. Click the connect button and you will get redirected to a new browser window with a command shell to the Red Hat OpenShift cluster

Boot node and bastion service on Microsoft Azure

When you start a Maximo Application Suite installation, a boot node is created that controls and completes the installation. In the Red Hat OpenShift cluster, a bastion service is configured to allow Secure Shell (SSH) access to cluster nodes.

During a Maximo Application Suite installation on Azure, virtual networks (VNet) are created that contain Azure virtual machines. For example, a VNet is created to contain the Red Hat OpenShift cluster, and virtual machines are created in the cluster to represent its master and worker nodes.

The boot node

In the Azure cloud, after you specify the installation parameters and start the installation, a boot node is created. The installation parameters are passed to the boot node. In addition, all of the required tools to complete the installation, such as Terraform and Docker, are installed on the boot node. By using these tools and parameters, the boot node performs the following tasks to complete the installation:

- Creates the virtual network infrastructure, such as the VNet that contains the Red Hat OpenShift cluster.
- Runs the bootstrap process that creates the Red Hat OpenShift cluster.
- Installs the Maximo Application Suite prerequisites.
- Installs Maximo Application Suite.
- Performs postinstallation validation.
- Stores the installation context and Terraform state files both locally and in the Azure Blob storage container that is associated with your Azure subscription.

Because it is located in its own VNet, the boot node is not part of the Maximo Application Suite Red Hat OpenShift cluster. After the installation is complete, you do not need to use the boot node to access the cluster or interact with Maximo Application Suite. For this reason, the boot node is kept in a shutdown state. However, if required, you can restart it and use it to troubleshoot installation issues.

The bastion service

Maximo Application Suite deployment uses the Azure Bastion service to enable the SSH access to the Red Hat OpenShift cluster nodes. This service is configured during the deployment and can be used to access the Red Hat OpenShift cluster nodes from the Azure portal. See the "Accessing the boot node and Red Hat OpenShift cluster" on page 84 section for the steps.

Note: No charges apply to Azure virtual machine that are in a shutdown state, such as the boot node. However, charges apply for their attached Premium SSD LRS Azure disk volumes of 30 GB. For more information, see Azure managed disk pricing in the Azure documentation.

IBM Maximo Application Suite installation with Ansible collection

To automate some of the manual steps that are involved with installing Maximo Application Suite and its components, use the Ansible collection roles that match your installation path or use case.

The IBM Maximo Application Suite development team maintains a public Ansible collection that automates the installation and configuration of Maximo Application Suite and its dependencies. The Maximo Application Suite DevOps Ansible collection includes a number of automated tasks, referred to as roles and playbooks. These automated tasks can be used to streamline processes from having a simple Red Hat OpenShift cluster, to having Maximo Application Suite including multiple applications and its required dependencies.

It also provides a docker container, which contains all the prerequisites to run the Ansible automation on any local machine.

For example, from Maximo Application Suite 8.9, you can use the role variable *mas_annotations* to install the Maximo Application Suite in production or nonproduction mode. The *mas_annotations* is an optional variable, which accepts a comma-separated list of annotations that need to be added to the Maximo Application Suite CR. To deploy your Maximo Application Suite in nonproduction mode, set this variable to

`mas.ibm.com/operationalMode=nonproduction`. For more information, see “ Maximo Application Suite Ansible collection examples” on page 88.

Note: The Maximo Application Suite Ansible collection is developed by the IBM Maximo Application Suite development team. If you need help or have issues, contact IBM Support or raise an issue directly in the GitHub repository.

For more information about how to help with the development of new roles and collections or improvements to the existing ones, see Contributing.

- Ansible role documentation contains terms and variables, prefixed by `W3` or `ARTIFACTORY` that are intended for internal IBM use only. Ignore these variables if they do not apply to your role, use-case, environment, or scenario.

Mapping documentation tasks to Ansible roles

Each documentation task that contains an Ansible role begins with a Tip that links to the Ansible role that applies to the task.

For reference, the following documentation tasks map to Ansible roles:

Documentation task	Ansible role
Setting up Red Hat OpenShift Container Storage	ocs
IBM operator catalog	ibm_catalogs
Installing IBM Cloud Pak foundational services for IBM Cloud Pak for Data	common_services
Ansible dev-ops role - cert_manager	cert_manager
Note: Starting in IBM Maximo Application Suite 9.0, 8.11.7, and 8.10.10, the User Data Services (UDS) is deprecated and replaced with IBM Data Reporter Operator (DRO). For more information, see Data Reporter Operator.	uds
IBM Data Reporter Operator	dro
Installing IBM Cloud Pak for Data	cp4d
Db2 Warehouse	db2
Watson Studio	cp4d_service Specify `wsl` service name.
Watson Machine Learning	cp4d_service Specify `wml` service name.
Installing MongoDB	mongodb
IBM Cloud Object Storage	cos Specify `cos_type` as `ibm` for IBM Cloud Object Storage, or `ocs` Red Hat OpenShift Container Storage.
Installing Suite License Service	sls
Recommended: Installing Suite Applications from Operator Hub	suite_app_install

Documentation task	Ansible role
Activating applications	suite_app_config
Installing Red Hat OpenShift Container Platform on IBM Cloud	ocp_provision
IBM Cloud Internet Services	suite_dns
Installing Apache Kafka for IBM Maximo Manage Installing Apache Kafka for IoT tool	kafka
Installing the NVIDIA operator	nvidia_gpu
Configuring Red Hat OpenShift cluster monitoring Installing Grafana	cluster_monitoring
Converting IBM Maximo Application Suite from manual deployment to channel subscription This Operator Lifecycle Manager (OLM) conversion script and role is available from Maximo Application Suite 8.10.	convert_to_olm

Maximo Application Suite Ansible collection examples

You can use Ansible collections to install Maximo Application Suite, its applications and prerequisites, and complete other related tasks. Review the examples to learn how to run the ansible roles and playbooks.

For more information, see Ansible DevOps documentation.

You can run Ansible playbooks and roles in two ways:

- Install prerequisite software and ensure it is available on your workstation. For more information, see MAS DevOps Ansible Collection.
- Use a Docker container that contains everything you need. For more information, see Docker container.

  Ensure that Docker is installed on your workstation.

The following examples use the Maximo Application Suite DevOps Docker container to run roles and playbooks. You set values for environment variables, then run the corresponding role or playbook.

Running the role to install the IBM Certificate Manager

1. In your local machine with Docker installed, run this command to pull the image and initiate the Docker container:

   ```
   docker run -ti --rm --pull always quay.io/ibmmas/cli
   ```

   Tip: Use the **`--pull always`** command to pull the newest image.

   Note: The **`--rm`** command ensures that the container is removed after you run the command. However, running the **`--rm`** command is optional.

   For more information, see ibmmas/cli.

1. In the command line inside the container, you can run the following commands:

   a. Run the **`oc login`** command. You can access your Red Hat OpenShift cluster by using the **oc** command directly from a terminal in the client machine **oc** was installed to.

      i) Go to the Red Hat OpenShift web console.

      ii) Click your login name and select the option: Copy login command.

iii) Click View token.

iv) Copy the entire command line under the Log in section with this token and paste in the command line, running it from inside the docker container.

b. Export the environment variable used by the script. In the example, IBM Certificate Manager is installed.

```
export MAS_CHANNEL=8.8.x
```

c. Run the cert_manager **role**.

```
ROLE_NAME=cert_manager ansible-playbook ibm.mas_devops.run_role
```

After several minutes, IBM Certificate Manager is installed.

Running a playbook to provision an IBM Cloud Red Hat OpenShift cluster

This example shows how to run a playbook to provision an IBM Cloud Red Hat OpenShift cluster.

Note: This example shows a playbook that just runs one role. The playbook sets the environment variables and the specific role. You might do the same using the steps that are shown in the previous example. But an advantage of playbooks is that you can run more than one role and also set the variables for all them as needed in the same file. In other words, the playbook is usually used to orchestrate an execution of a set of roles.

For example, a playbook can install Maximo Application Suite and perform more configurations after it is installed. For more information, see suite_install.

1. In your local machine with Docker installed, run this command to pull the image and initiate the Docker container:

```
docker run -ti --rm --pull always quay.io/ibmmas/cli
```

2. In the command line inside the Docker container, run the following commands:

a. Update `ibmcloud cli`, including plug-ins, to the newest version.

```
curl -sL https://raw.githubusercontent.com/IBM-Cloud/ibm-cloud-developer-tools/master/
linux-installer/idt-installer | bash
```

b. Now, export environment variables necessary to provision your IBM Cloud Red Hat OpenShift cluster.

Note: For more information about variables to customize your IBM Cloud Red Hat OpenShift cluster, see Role Variables - ROKS.

c. Export your IBM Cloud API Key.

```
export IBMCLOUD_APIKEY=<your IBM Cloud API Key>
```

Note: To create a key, see Creating your IBM Cloud API key

d. Export the name of your cluster.

```
export CLUSTER_NAME=<my own cluster>
```

e. Run the command to run the playbook:

```
ansible-playbook ibm.mas_devops.ocp_roks_provision.yml
```

3. The IBM Cloud cluster is provisioned after some time.

Customizing a playbook to run existent roles

In this example, customize a playbook to run a set of roles. The playbook installs IBM catalog, IBM Cloud Pak for Data, and IBM User Data Services, which is a Maximo Application Suite prerequisite.

To perform the task, the playbook sets the variables and runs the specific roles to install each one of them.

1. In your local machine with Docker installed, run this command to pull the image and initiate the Docker container:

```
docker run -ti --rm --pull always quay.io/ibmmas/cli
```

2. In the command line inside the container, you can run the following commands:

   a. Run the `oc login` command. You can access your Red Hat OpenShift cluster by using the `oc` command directly from a terminal in the client machine `oc` was installed to.

      i) Go to the Red Hat OpenShift web console.

      ii) Click your login name and select the option: Copy login command.

      iii) Click View token.

      iv) Copy the entire command line under the Log in section with this token and paste in the command line, running it from inside the docker container.

3. Create a `custom_dro_playbook.yaml` file with the following content. (You can use an editor such as vi and insert the following sample code.)

```
...
    - hosts: localhost
      any_errors_fatal: true
      vars:
        dro_contact:
          email: "{{ lookup('env', 'DRO_CONTACT_EMAIL') }}"
          first_name: "{{ lookup('env', 'DRO_CONTACT_FIRSTNAME') }}"
          last_name: "{{ lookup('env', 'DRO_CONTACT_LASTNAME') }}"
      roles:
        # 1. Install DRO
        - ibm.mas_devops.ibm_catalogs
        - ibm.mas_devops.common_services
        - ibm.mas_devops.dro
```

4. Export the required environment variables to install Data Reporter Operator .

```
export DRO_CONTACT_EMAIL=john.doe@test.com
```

```
export DRO_CONTACT_FIRSTNAME=John
```

```
export DRO_CONTACT_LASTNAME=Doe
```

5. Run the playbook with the following command:

```
ansible-playbook custom_dro_playbook.yaml
```

6. Data Reporter Operator was installed along with its prerequisites.

Installing on nonproduction environment

This example describes how to install the IBM Maximo Application Suite in production or nonproduction environment by using the *mas_annotations* role. The *mas_annotations* is an optional variable, which accepts a comma separated list of annotations that need to be added to the Maximo Application Suite CR.

1. In your local machine with Docker installed, run a command to pull the image and initiate the Docker container:

```
docker run -ti --rm --pull always quay.io/ibmmas/cli
```

2. In the command line, inside the Docker container, run the following command to deploy the Maximo Application Suite in nonproduction mode for development and testing deployment.

   Set environment variable MAS_ANNOTATIONS for nonproduction mode:

```
export MAS_ANNOTATIONS=mas.ibm.com/operationalMode=nonproduction
```

3. Run the following command to install the Maximo Application Suite:

```
...
- hosts: localhost
  any_errors_fatal: true
  vars:
    mas_instance_id: "inst1"
    mas_config_dir: "/home/david/masconfig"
    mas_entitlement_key: "{{ lookup('env', 'IBM_ENTITLEMENT_KEY') }}"
    mas_annotations: "mas.ibm.com/operationalMode=nonproduction"

  roles:
    - ibm.mas_devops.suite_install
    - ibm.mas_devops.suite_config
    - ibm.mas_devops.suite_verify
```

Setting up IBM Maximo Application Suite

After you install IBM Maximo Application Suite, the setup program guides you through the initial configuration.

Before you begin

1. Complete the installation.

   Obtain the link to the Maximo Application Suite setup program and the login credentials that you need to complete the setup process.

   For more information about obtaining the login credentials, see how to locate the default username and password.

2. Enable login for Maximo Application Suite self-signed certificates.

   If you are using self-signed certificates in a development or test environment, you must manually enable login by using either of the following methods.

   - Download the certificates from the cluster and add them to your local certificate manager.
   - In your browser, go to the Maximo Application Suite API URL `https://api.<mas_domain>/` and accept the certificate security risks. After you accept the risks, an `AIUC01999E` error is displayed. This message is expected. You can now continue with the setup process.

   If the Maximo Application Suite dashboard does not load after you login for the first time and instead see a spinning wheel, see how to troubleshoot the issue.

About this task

The Maximo Application Suite setup configurations are set at the System scope. For more information about configuration scopes, see Configure Maximo Application Suite.

Procedure

1. Log in to the Maximo Application Suite setup program by using the superuser credentials that were created during installation.

`https://admin.<mas_domain>/initialsetup`

Important: Treat the superuser account the same way that you treat the root account on your servers. Use it only for the initial setup. As part of the setup, you create a default administrator user account that has access to the Maximo Application Suite administrative interface. Use this administrative account to add and manage users, deploy applications, and more.

For more information about obtaining the superuser credentials, see how to locate the default username and password.

2. Configure MongoDB.

   MongoDB is used as the data dictionary for Maximo Application Suite and its applications. It is also used as the default user registry.

   Specify the following MongoDB information:

   Hostname and port
   You can configure one or more MongoDB hostname and port combinations.

   Authentication mechanism
   Specify the mechanism that is used to authenticate Maximo Application Suite when it connects to MongoDB. Select the closest match to the mechanism that is configured for your MongoDB cluster. For example, if your cluster uses the SCRAM-SHA-256 mechanism, select **DEFAULT (SCRAM)**.

   To authenticate by using LDAP, specify **PLAIN** as the authentication mechanism.

   Auth db

   Provide the name of the authentication database. If you are authenticating with LDAP, the value must be `$external`.

   MongoDB login credentials
   At a minimum, the MongoDB administrator needs table creation privileges.

   Note: The MongoDB verification might take up to a minute. The configuration cannot be modified after the MongoDB verification is complete. MongoDB is a prerequisite for Maximo Application Suite. Changing the configuration requires careful coordination and possible data migration to avoid service outages. System administrators can change the configuration in the Red Hat OpenShift console. For assistance with changing the MongoDB configuration, contact your IBM representative.

   For more information, see Installing MongoDB.

3. Upload a CA certificate.

   If the service uses the transport layer security (TLS) communication protocol and is not secured with a certificate that is issued by a well-known certificate authority (CA), then provide the certificate of the CA that issued the service's certificate. Because the CA might use intermediate CAs, you can provide more than one certificate.

   For each certificate that you provide, the following details are displayed:

   - The name of the certificate issuer.
   - The name of the subject, such as the organization, that the certificate is issued to.
   - The start and end dates of the certificate's validity period. If the validity of any certificate that you provide expires soon, a warning message appears.

   You can automatically retrieve or manually add certificates.

   Important: If your MongoDB cluster uses self-signed CA certificates that you must retrieve or add a certificate.

   - Automatically retrieving certificates

     In the certificates section, click **Retrieve**. If the connection credentials that you specify are correct, all CA certificates that are configured on the server are automatically retrieved and displayed.

     These certificates are not validated. Verify that only the correct certificates are retrieved and remove any unexpected certificates.

After you retrieve the certificates, you can manually add more certificates.

- Manually adding certificates

  In the certificates section, click **Add manually** and specify the following values for each certificate that you want to add:

  Alias
  An alphanumeric identifier that is in the range 3 - 50 characters long.

  Certificate content
  The content of a certificate file in either the X.509 or PEM formats.

For more information, see Configuring certificate authority certificates.

4. Configure a Simple Mail Transfer Protocol (SMTP) server connection to enable email notifications for system events, such as new user welcome emails and password reset communication. For more information, see Setting up email notifications.
5. Configure analytics data.

   Note: Starting in IBM Maximo Application Suite 9.0, 8.11.7, and 8.10.10, the User Data Services (UDS) is deprecated and replaced with IBM Data Reporter Operator (DRO).

   For more information, see Data Reporter Operator.

   - If you are using Maximo Application Suite 8.11, 8.10 or earlier versions, you must migrate your User Data Services to Data Reporter Operator . For more information, see Migrating Maximo Application Suite from User Data Services to Data Reporter Operator.
   - If you are using Maximo Application Suite 9.0, 8.11.7, 8.10.10 or later versions, configure IBM Data Reporter Operator.

     The IBM Data Reporter Operator accepts events and transforms them into reports that are submitted to the Data Service of the IBM Metrics Operator.

     a. Enter the following information to configure Data Reporter Operator for Maximo Application Suite:

        – **URL** - This URL is the DRO URL endpoint. To find it, go to your Red Hat OpenShift console, switch to ibm-common-services project, then **Networking** > **Routes**. Copy the URL displayed under the Location column for the dro-endpoint route.

          For example, `https://dro-endpoint-ibm-common-services.<your-cluster-domain>`

        – **API Key** - This API key is the DRO API Key credential. To find it, go to you Red Hat OpenShift console, switch to `ibm-common-services` project, then **Workloads** > **Secrets** > **Search and select the secret named dro-api-key**. Under the **Data** section, copy the apikey value.

          For example, `k2wnQY...`

        – **Email** - Enter a contact email address to use for DRO communication. The email address does not have to match an existing Maximo Application Suite user.

        – **Given Name** - Enter the given name of the owner of the provided contact email address that is used for DRO communication.

        – **Surname** - Enter the surname of the owner of the provided contact email address that is used for User Data Services communication.

        – **Certificates -** Enter the chain of SSL certificates for your DRO. To retrieve the certificates, you can click the **Retrieve button (under Certificates section)** while configuring DRO into Maximo Application Suite. The DRO certificates to configure in Maximo Application Suite will vary according to the cloud service provider's cluster that is hosting your DRO installation.

     b. Click **Add** to add the **intermediate of the certificate chain**.

     c. Enter an **alias**. **Example**: drocertpart1.

d. Enter the **Certificate content.**. Include **the Let's Encrypt R3 intermediate certificate**, issued to **US, Let's Encrypt, R3**. For more information, see certificate content. Example:

```
-----BEGIN CERTIFICATE-----
MIIF5jCCBM6gAwIBAgISA0Y...
-----END CERTIFICATE-----
```

e. Click **Confirm**. The first part of this certificate should include valid dates and look like the following example:

```
Issued to: US, Let's Encrypt, R3
Issued by: US, Internet Security Research Group, ISRG Root X1
Valid from: Thu Aug 01 2024
Valid to: Mon Sep 15 2025

This is the intermediate certificate which is required for the SSL connection to DRO
endpoint.
```

f. Click **Add** to add the **root of the certificate chain**.

g. Enter an **alias**. **Example**: drocertpart2.

h. Enter the **Certificate content**. Include the ISRG Root X1 self-signed certificate. For more information, see certificate content. Example:

```
-----BEGIN CERTIFICATE-----
MIIFazCCA1OgAw...
-----END CERTIFICATE-----
```

i. Click **Confirm**. The **second part of this certificate** should have valid dates and look like the following example:

```
Issued to: US, Internet Security Research Group, ISRG Root X1
Issued by: US, Internet Security Research Group, ISRG Root X1
Valid from: Thu Jun 04 2015
Valid to: Mon Jun 04 2035


This is the root certificate which is required for the SSL connection to DRO endpoint.
```

j. **Save** the DRO configuration.

k. Now, wait for the DRO configuration to reconcile, this process might take up to 10 minutes. The configuration will be successfully completed when the configuration status is set to Ready. Example:

```
Configuration Ready - DRO configuration was successfully verified
```

6. Configure the Suite License Service.

   The Suite License Service (SLS) stores and manages the Maximo Application Suite license.

   Each Maximo Application Suite instance can be connected to a unique SLS instance. Two or more Maximo Application Suite instances can also share an SLS and the corresponding license file.

   Enter the following SLS information to configure Maximo Application Suite:

   - URL - The URL for the SLS server.
   - Registration key - Enter the SLS registration key.

   Depending on your environment, the SLS configuration might take 10 minutes or more to complete.

7. Optional: Upload your license key file.

   If the IBM Suite License Service that you configured for use with Maximo Application Suite includes a valid license file, you do not need to upload a license file. You can continue with the next configuration step.

To activate Maximo Application Suite, you must provide your license key from the IBM License Key Center. The login information is provided in the license Key Center welcome letter. For more help on licensing, see the IBM Support - Licensing page.

a) Log in to the license Key Center.

b) Select your company name.

c) Select the **IBM AppPoints** product line.

d) Select the `IBM Maximo Application Suite...` license key name.

e) Select the product or sales order for which to create the license key.

f) Enter the number of keys to generate. These correspond to the AppPoints that are allocated to the license key.

g) Provide the Maximo Application Suite license server parameters.

Use the parameters that are displayed in the **Advanced settings** > **license key** section of the Maximo Application Suite setup program, or provide the following parameters:

- For Configuration, specify a Single License Server.
- For Host ID type, specify the Ethernet address.
- For Host ID, specify the host ID that was generated when you installed the Suite License Service (SLS). To display this ID, connect to your Red Hat OpenShift cluster and run the following command:

  ```
  oc -n <sls_project_namespace> get licenseservice sls
  ```

  For example, if the namespace of the SLS project is `mas-sls-dev5`, run the following command:

  ```
  oc -n mas-sls-dev5 get licenseservice sls
  ```

  In the command output, the host ID is displayed in the `LICENSEID` column.
- For Hostname, specify a hostname of your choice, for example: `sls-mas`
- For Port, specify 27000.

h) Download the key and then upload it to the Maximo Application Suite setup program.

8. Create the workspace.

The Maximo Application Suite workspace is a unique collection of configuration settings for your instance of Maximo Application Suite. Enter the following information to create your Maximo Application Suite workspace:

- Workspace ID

  The workspace ID forms part of the Maximo Application Suite URL, for example:

  `https://<workspace_id>.home.<mas_domain>`

  Note: The workspace ID must be 3 - 12 characters in length, and can contain only lowercase letters and numbers. The first character must be a letter.
- Workspace display name

  The display name is shown in your Maximo Application Suite user interface.

9. Review the setup configuration.

Your Maximo Application Suite setup is now complete. Verify that all configuration settings are done and then click **Finish** to complete the setup.

What to do next

After the Maximo Application Suite setup is complete, you can start to use your environment by going to the Maximo Application Suite administration or the Maximo Application Suite navigator page:

```
https://admin.<mas_domain>
https://<workspace_id>.home.<mas_domain>
```

As the Maximo Application Suite superuser, you can now continue configuring your environment to suite your enterprise needs:

- Configure authentication

  Maximo Application Suite supports local user authentication by MongoDB and authentication by using LDAP or SAML.
- Configure LDAP user registry synchronization

  User registry synchronization simplifies Maximo Application Suite user management by synchronizing users and groups between an LDAP server and your local Maximo Application Suite user registry.
- Create administrator user accounts

  The initial superuser account is used to complete the Maximo Application Suite setup. You can add application administrator users or system administrator users for day-to-day administrative tasks.
- Getting started

  With the setup completed, your users can log in and start to use Maximo Application Suite.

Uninstalling

The method that you use to uninstall IBM Maximo Application Suite depends on your environment.

Customer-managed Uninstalling Maximo Application Suite

Starting in 8.10, you can uninstall the IBM Maximo Application Suite with the **`mas uninstall`** command by using the CLI command utility. By uninstalling Maximo Application Suite, you remove the core application and all deployed applications from your environment.

About this task

The uninstallation process sequentially removes the Maximo Application Suite applications, industry solutions, add-ons, and tools that were installed.

- After the industry solutions, add-ons, and tools are removed, all config maps and secrets are also removed.
- If you use Strimzi as the Kafka operator, the Kafka topics can be removed if needed.
- All entries that are related to the instance can be removed from the MongoDB instance if needed. During this phase, double confirmation is required for each database removal to ensure no unexpected data loss. A confirmation string is created at the start of this phase to provide the confirmation for each removal.
- The default ClusterIssuer resource is also removed if one was not provided on the installation. If you created and provided your own ClusterIssuer resource to the installation script, it is not removed.
- Finally, all projects and namespaces that are related to this instance are removed.

Even though multiple instances of Maximo Application Suite might be installed on a single cluster, the uninstaller removes just one instance at a time. Run the uninstall multiple times to remove each instance.

Note: Supporting components, such as MongoDB and IBM Cloud Pak for Data, are not removed by the uninstaller and must be removed separately if needed.

You can also automate the Maximo Application Suite uninstallation. First run the ansible role `suite_app_uninstall` to remove the applications and then use the ansible uninstall core playbook to uninstall Maximo Application Suite core platform and its dependencies from your cluster. For more information, see suite_app_uninstall role and uninstall core playbook.

Procedure

Run the **mas uninstall** [options] command either in interactive or non interactive mode.

1. Run the command in interactive mode.

   ```
   mas uninstall -i|--id MAS_INSTANCE_ID
   ```

   where,

   -i refers to interactive mode

   --id *MAS_INSTANCE_ID* refers to the Maximo Application Suite that is to be uninstalled

   Other options include **--no-confirm** to launch the uninstall without prompting for confirmation and **-h|--help** to display a help message

   Alternatively, you can run the command in non interactive mode.

   ```
   mas uninstall -i MAS_INSTANCE_ID -\-no-confirm
   ```

2. If you are not connected to a Red Hat OpenShift cluster, you are prompted to provide the server URL and token, and whether to verify the server certificate or not.

   If you are already connected to a cluster you can opt to change to another cluster.

Deleting the Maximo Application Suite stack on Amazon Web Services

Download and run a script that deletes the IBM Maximo Application Suite stack, virtual infrastructure, VPCs, and EC2 instances from your Amazon Web Services (AWS) account.

When you install an instance of the Maximo Application Suite, several virtual infrastructure resources are created in your AWS account, such as virtual private clouds (VPC), Amazon EC2 instances, and a CloudFormation stack. To uninstall the Maximo Application Suite, you download and run a script that deletes these resources. You can use this script to uninstall the Maximo Application Suite regardless of whether the installation succeeded or failed.

Attention: Do not delete the CloudFormation stack before running the script to uninstall Maximo Application Suite. The script deletes the stack. If the stack is deleted manually, the uninstallation might fail.

You run the script on your local machine or on a server that is not located in a VPC that the installation process created, such as the VPC that contains the Red Hat OpenShift cluster.

Note: Do not run the script in any of the EC2 instances that the Maximo Application Suite installation process created, such as the Bootnode, the bastion host, or any of the cluster nodes. The script deletes these EC2 instances. If you run it in any of them, it fails.

Before you begin

- On the machine where you want to run the script, ensure that the following CLI packages are installed:
  - Version 4.0 or a later version of GNU bash
  - jq
  - AWS CLI

    Ensure that the AWS CLI package is configured for authentication with your AWS account. For more information, see Configuring the AWS CLI in the AWS documentation.

- If Amazon DocumentDB or Amazon MSK are configured in the Maximo Application Suite stack, you must delete the Amazon DocumentDB or Amazon MSK instance.

  For more information, see Deleting an Amazon DocumentDB cluster or Deleting an Amazon MSK cluster.

Procedure

1. In a browser window, open the script, right-click the page and save the script to your local machine by using the name `cleanup-mas-deployment.sh`.
   a. If you do not want to run the script locally, use SCP or any file transfer tool to copy the script to the machine where you want to run it.
   b. On the machine where you want to run the script, in a command shell, log in to the AWS service by running the following command:

      ```
      aws configure
      ```

      You are prompted for your identity and access management (IAM) user credentials. Enter the credentials for an IAM user that has the permissions to run the script, such as the IAM user that installed the Maximo Application Suite. For more information, see Configuring the installation permissions.
   c. Make the script executable by entering the following command:

      ```
      chmod +x cleanup-mas-deployment.sh
      ```

   d. View the script's usage information by running the following command:

      ```
      ./cleanup-mas-deployment.sh -h
      ```

2. Run the script.
   - You must specify the region code of the region where the Maximo Application Suite was installed by using the `-r` option, for example: `-r ap-northeast-3`
   - To delete the virtual resources by using the CloudFormation stack name, use the `-s` option.

   For example, if the CloudFormation stack name is `sp-manage-12` and the region code is `ap-northeast-3`, run the following command:

   ```
   ./cleanup-mas-deployment.sh -s sp-manage-12 -r ap-northeast-3
   ```

3. Verify that the script completed successfully.

   If the script is successful, output that is similar to the following text is displayed:

   ```
   $ ./cleanup-mas-deployment.sh -s sp-manage-12 -r ap-northeast-3
   Stack name: sp-manage-12
   Unique string:
   Region: ap-northeast-3
   Supported region provided
   Deleting by stack-name sp-manage-12
   Execution started at Mon Mar  7 22:46:48 IST 2022
   MAS instance unique string: nove9h
   Checking for EC2 instances
   …
   EC2 instances found for this MAS instance
   …
   Terminate request submitted
   Waiting for instances to be terminated
   Deleted EC2 instances
   Checking for volumes
   …
   Found volumes for this MAS instance
   …
   Checking for VPC
   VPC_ID = vpc-0851c8fc0523cac86
   Found VPC with Id vpc-0851c8fc0523cac86 for this MAS instance, it will be deleted at the end
   Checking for NAT gateways
   …
   Found NAT gateways for this MAS instance
   ```

```
…
Checking for EIPs
…
Checking for load balancers
…
Checking for v2 load balancers
…
Checking for network interfaces
…
Checking for internet gateways
…
Checking for subnets
…
Checking for routing tables
…
Checking for network ACLs
…
Checking for security groups
…
Checking for S3 buckets
…
Checking for IAM users
…
Checking for IAM instance profiles
…
Checking for IAM policies
…
Checking for IAM roles
…
Checking for private hosted zones
…
Checking for CloudWatch log groups
…
Checking for CloudFormation stack
…
Execution completed at Mon Mar  7 23:02:47 IST 2022
```

4. In the AWS CloudFormation console, verify that the stack that you created when you installed the Maximo Application Suite is deleted.
5. In the AWS VPC console, verify that the virtual infrastructure that was created when the Maximo Application Suite was installed is deleted.
6. Verify that no VPCs exist that contain `<unique-string>` in the VPC name, for example: `masocp-<unique-string>-vpc`
7. Verify that no EC2 instances exist that contain `<unique-string>` in the EC2 instance name.

   For more information on `<unique-string>` and other identifiers that are used in this documentation, see Unique identifiers.

Uninstalling Maximo Application Suite on Microsoft Azure

To uninstall IBM Maximo Application Suite on Microsoft Azure, all the infrastructure resources that are created during the deployment must be deleted. Additionally, if a paid product is installed, unsubscribe the product from your Microsoft Azure account.

When you install an instance of the Maximo Application Suite, several virtual infrastructure resources are created in your Microsoft Azure account. These resources include virtual network (VNet), Microsoft Azure virtual machines, storage, and other resources. To uninstall the Maximo Application Suite, you download and run a script that deletes these resources. You can use this script to uninstall the Maximo Application Suite regardless of whether the installation succeeded or failed.

You run the script on your local computer or on a server that is not located in a VNet that the installation process created, such as the VNet that contains the boot node or the Red Hat OpenShift cluster.

Note: Do not run the script in any of the virtual machines that the Maximo Application Suite installation process created, such as the boot node, or any of the cluster nodes. The script deletes these virtual machines. If you run it in any of them, it fails.

Before you begin

On the computer or server where you want to run the script, ensure that the following CLI packages are installed:

- GNU bash version 4.0 or later
- jq
- Microsoft Azure CLI

  Ensure that this package is configured for authentication with your Microsoft Azure account. For more information, see Configuring the Azure CLI in the Microsoft Azure documentation.

Procedure

Uninstall Maximo Application Suite depending on your product type.

1. Delete the infrastructure resources.

   This step is applicable to BYOL and paid products.

   a) In a browser window, open the script, right-click the page and save the script to your local machine by using the name `cleanup-mas-deployment.sh`.

   b) If you do not want to run the script locally, use SmartCloud Provisioning, or any file transfer tool to copy the script to the machine where you want to run it.

   c) On the machine where you want to run the script, in a command shell, log in to the Microsoft Azure service by running the following command.

   ```
   az login
   ```

   A browser opens where you can log in with the Microsoft Azure credentials. After login, you can close the browser window and continue the next steps from the command shell.

   d) Run the following command:

   ```
   chmod +x cleanup-mas-deployment.sh
   ```

   e) View the script's usage information by running the following command:

   ```
   ./cleanup-mas-deployment.sh -h
   ```

   f) Specify your preferred options and run the script.

   You must specify either the boot node resource group where the Maximo Application Suite was installed by using the `-r` option, for example: `-r mas-ocp-deploy-rg` or the unique string that is associated with the Red Hat OpenShift cluster, for example: `-u nove9h`

   To delete the virtual resources by using the resource group name, use the `-r` option. For example, if the boot node resource group name is `mas-ocp-deploy-rg`, run the following command:

   ```
   ./cleanup-mas-deployment.sh -r mas-ocp-deploy-rg
   ```

   g) To delete the virtual resources by using the installation identifier that is <`unique-string`>, use the `-u` option.
   For example, if <`unique-string`> is `nove9h`, run the following command:

   ```
   ./cleanup-mas-deployment.sh -u nove9h
   ```

   The script takes 15 - 20 minutes to delete the cluster and the virtual network infrastructure.

   h) Verify that the script completed successfully. If the script is successful, output that is similar to the following text is displayed:

   ```
   $ ./cleanup-mas-deployment.sh -r mas-ocp-deploy-rg
   ==== Execution started at Tue May 10 13:17:10 EDT 2022 ====
   Script Inputs:
    Bootnode resource group = test-mas-1
    Unique string =
   SUB_ID: b2ca5467-2502-4b05-b78e-744604c6531d
   Trying to delete OCP cluster resource group
   Deleting by 'bootnode-resource-group' test-mas-1
   UNIQ_STR: vt57ov
   ```

```
…
==== Execution completed at Tue May 10 13:26:53 EDT 2022 ====
```

i) On the Microsoft Azure portal in the Resource group, verify that both the resource groups (boot node resource group and Red Hat OpenShift cluster resource group) are deleted.

2. Unsubscribe the paid product.

   This step is applicable to the paid product only.

   For more information about canceling the Maximo Application Suite subscription, see https://learn.microsoft.com/en-us/azure/cost-management-billing/manage/cancel-azure-subscription.

Deploying Maximo Manage

You deploy Maximo Manage or Maximo Manage with Maximo Health operator, which is responsible to control the deployment process and to maintain and update the application as necessary.

Before you begin

- You also can check the actual supported matrix of your current Maximo Manage application that is deployed and the matrix of compatibility between the components, through the Red Hat OpenShift console:

1. Go to the Red Hat OpenShift console, in the Workloads/Pods section, select the namespace of your Maximo Manage instance in Projects, for example, `mas-<yourmasinstancename>-manage`, and then click the `ibm-mas-manage-operator-somestring` pod.
2. On the **Terminal** tab, select to connect to **webhook** container.
3. Open the supported matrix JSON file that displays the components version that is supported by each Manage application version and look for the version that you installed. If you are not sure what the version is, click **Details** and click **Manager** in the **Containers** section. The version is displayed in the **lmage version** tab or field. When you go back to the terminal, you can use this command to open the file:

   `cat /manage-admission/metadata/supported-versions-matrix.json`

   The file has a format like this:

```
  "8.3.1": [
    {
      "name": "anywhere",
      "versions": [
        "8.0.1",
        "8.0.0"
      ]
    }
```

   Note: In this example, the Manage application version is 8.3.1. The following list shows the Anywhere versions that are supported by this version.

   When you select **latest**, the page displays a section that states the current Manage application version:

```
"latest": {
    "base": "8.3.1",
    "anywhere": "8.0.1"
```

4. Open the dependency matrix JSON file that displays the compatibility between each component by using the following command: **`cat dependency-matrix.json`** The file displays the components with the following attributes:

```
"aviation": {
                "description": "Maximo Aviation",
                "includesForbidden": [
                        "hse",
                        "serviceprovider"
```

```
                ],
                "includesCoexist": [
                        "acm",
                        "transportation"
                ],
                "conflict": [
                        "civil",
                        "health",
                        "oilandgas",
                        "oracleadapter",
                        "nuclear",
                        "sapadapter",
                        "spatial",
                        "utilities"
```

Note: **`includesForbidden`** The component cannot be co-deployed with the listed components.
`includesCoexist` The component can be co-deployed with the listed components.
`conflict` The component cannot be co-deployed with the listed components.

Procedure

1. From the Suite catalog, on the **Applications** tab, select the **Manage** with tile, and review the information on the **Setup** page.
   a) Optional: If you plan to co-deploy industry solutions or add-ons with Manage, select the **Show all** option and then click the plus icon on the respective tile of each industry solution or add-on that you want to co-deploy to reserve the necessary AppPoints for them.
   b) Click **Continue**.
2. In the **Administer application upgrades** window, in the **Upgrade strategy** field, select one of the following upgrade strategies.

Option	Description
Channel subscription	a. To make sure that updates are automatically offered when an updated version is added to a channel, select **Channel subscription**. You can choose whether this update occurs automatically with a new version or requires manual approval. If you choose an automatic upgrade strategy through **Channel subscription**, required downtime might occur before you have a chance to review changes in the new version or fix pack, back up the database, or take other preparatory action. Therefore, for production Maximo Manage deployments, set the approval mode to **Manual**. You can then be alerted about the availability of an update directly in the IBM Maximo Application Suite application catalog, in the Maximo Manage tile. However, the update is not started until you trigger it yourself. In this case, you can be better prepared by reviewing the changes, running backups of the Maximo Manage configuration and custom resource definitions, scheduling the update, and communicating the scheduled downtime to users. b. In the **Channel** field, select a channel. c. In the **Custom source** field, specify a source. d. In the Channel details section, select whether to approve the update automatically or to require manual approval. e. Click **Subscribe to channel**.
Manual	a. To manually update the application when you are notified that an updated version is available, select **Manual**. b. In the **Version** field, select a version. c. Click **Deploy version**.

3. Configure the database connection information for Maximo Manage.
   a) In the Integrations and dependencies section, on the **Database connection** tile, select **Configure**.

b) On the **Database connection** page, click **Configure**.

c) In the **JDBC connection information** section, specify the following fields:

Note:

If you are upgrading from Maximo Asset Management to Maximo Application Suite, refer to the `maximo.properties` file of your Maximo Asset Management folder to get the values for hostname, port, database name to use in the jdbc url as required by Maximo Manage.

i) If you want to use an SSL-enabled connection, specify this field by using one of the following JDBC URL formats. Ensure that the **Port** that is used contains the SSL-enabled port of the database.

Oracle Database

TCPS is the protocol to use for Oracle SSL connections. For Oracle SSL database connections in Maximo Manage, you must specify `SID=<TNS Service ID>`. You can use the following URL as an example:

```
jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS=(PROTOCOL=TCPS)(Host=mymaximodb.com)
(Port=2484))(CONNECT_DATA=(SID=MAXDB)))
```

Microsoft SQL Server Database

For SQL Server SSL database connections in Maximo Manage, you must specify `encrypt=true`. Ensure that you use a semicolon to end the JDBC connection string. You can use the following URL as an example:

```
jdbc:sqlserver://mymaximodb.com:1433;databaseName=MAXDB;encrypt=false;
```

IBM Db2 database

For Db2 SSL database connections in Maximo Manage, you must specify `sslConnection=true`. Ensure that you use a semicolon to end the JDBC connection string. You can use the following URL as an example:

```
jdbc:db2://mymaximodb.com:50001/MAXDB:sslConnection=true;
```

a) In the **User name** field, specify the database username.

b) In the **Password** field, specify the database user password.

c) Ensure that you select the **SSL Enabled** option.

ii) If you want to use non-SSL enabled connection, specify the **Connection string** field by using one of the following JDBC URL formats, depending on the database you are using.

Oracle Database

You can use the following URL as an example:

```
jdbc:oracle:thin:@mymaximodb.com:1521:MAXDB
```

Microsoft SQL Server database

You can use the following URL as an example:

```
jdbc:sqlserver://
mymaximodb.com:1433;databaseName=MAXDB;integratedSecurity=false;encrypt=false;
```

IBM Db2 database

You can use the following URL as an example:

```
jdbc:db2://mymaximodb.com:50001/MAXDB
```

a) In the **User name** field, specify the database username.

b) In the **Password** field, specify the database user password.

c) Ensure that you do not select the **SSL Enabled** option.

d) In the Additional driver options section, in the **Driver options** field, add more driver options, which are separated by a semicolon.

- Typically, you can specify JDBC options as part of the URL for the database. However, in some cases you might want to specify JDBC options in the **Driver options** field.

For example, your URL might exceed the maximum length that is allowed, or you might want to configure a JDBC option that cannot be included in the connection URL. You cannot specify the same JDBC option in both the URL and the **Driver options** field. If you do, JDBC driver errors might cause the connection to fail.

- If you specify an extra JDBC option for your database, the CustomProxyDriver acts as a proxy driver that routes the database requests to the actual driver for your type of database.

e) If you chose to use an SSL-enabled database connection, in the Security > Certificates (optional) section, click **Add+** to display the fields to include in your database certificate.

i) In the **Alias** field, specify an alias name to identify the certificate, for example, DB2WHcert.

ii) In the **Certificate content** field, copy and paste your certificate in the format that is mentioned in the field content. You can retrieve a PEM certificate for your database. The file must be a Base-64 encoded X.509 file. You do not need to retrieve a private key. For more information, see the documentation for your database. After you copy and paste the text into the field, including the BEGIN CERTIFICATE, and END CERTIFICATE text, click **Confirm**.

f) Click **Save**. The **Database connection** page is closed.

g) Click in the **Database connection** tile to verify your database connection. Expand the **Status** icon that is loading in the Configuration-scope-Workspace-application section to display some tiles. Click **Select** once the **Status** icon is ready to close the page.

Note: Click **Save and Select** in the previous step if you do not want to wait for the database connection verification after you complete the fields.

h) If you plan to deploy Maximo Optimizer, see Deploying Maximo Optimizer.

4. In the **Components** section, select the industry solutions or add-ons and the version that you are also activating with Maximo Manage.

- If you select **latest** from the **New version** list for the selected industry solution or add-on, the version of the component that is supported by the current Maximo Manage application version is co-deployed with Maximo Manage base component.

Note: If your configurations settings were set to the latest for the components before you upgrade the application, the components will be automatically updated after the upgrade. If you select a specific version instead, then the components are not updated until you change their versions to the current, supported version of the new application version that was now updated, both by selecting the exact version or by selecting **latest**.

- If you select one component that is going to be deployed by selecting the **latest** option, you must select **latest** for any other component you co-deploy. If you select one component that is going to be deployed with an exact version number, you must select the exact version number for any other extra component that you co-deploy. You cannot mix exact version numbers and the latest version in the components you want to co-deploy.
- You also can click in the **New version** column for a component and select **Select version**. In the **Select unlisted version** dialog, you can specify a valid version that is supported by the current application version that is deployed in the **Version** field and click **Save**.

For more information, see Deployment of industry solutions and add-ons section, including access to the compatibility matrix that shows the compatibility between application version and components versions. Some components might not be compatible with each other.

5. Click **Show advanced settings** to view and specify the configuration settings, such as database, server bundle, language settings, and others.

a) In the Database section, clear the **System managed** checkbox and manually configure the database.

Schema
Enter the name of the schema that is configured in your database. For more database configuration information for Maximo Manage, see Setting up your database.

Encryption secret (optional)
This value is optional if you are deploying Maximo Manage in your database for the first time, and your database is not encrypted. Enter your encryption keys for this parameter. For more encryption settings information, see Database encryption.

In the Key/Value table, click **Add property +**. In the Key column, enter `MXE_SECURITY_CRYPTOX_KEY` and in the Value column, enter your encryption key value.

In the Key/Value table, click **Add property +**. In the Key column, enter `MXE_SECURITY_CRYPTO_KEY` and in the Value column, enter your encryption key value.

Table space
If the default value does not match your database configuration, enter the name of the table space that was configured in your database. For more database configuration information for Maximo Manage, see Setting up your database.

Index space
If the default value does not match your database configuration, enter the name of the index table space that was configured in your database. For more database configuration information for Maximo Manage, see Setting up your database.

Install demo data
If you are deploying a test or demonstration environment for Maximo Manage, you can install sample data.

The sample data in the demo database is useful for development or test environments.

To set up a test or development environment with demo data, install an instance of Maximo Application Suite specifically for testing or development. Then, when you configure the database settings for your Maximo Manage deployment, select the option to install demo data.

Note: You cannot add sample data after Maximo Manage is deployed because the database is updated without sample data. To add the sample data after deploying Maximo Manage, you must re-create or clean your database, and reconfigure Maximo Manage.

Db2 Vargraphic
If you use Db2 and you plan to install a language other than English for your base language or as an extra language, select this option. If you intend to add more languages later, select this option during your initial deployment. This option does not affect the Maximo Manage deployment if it is selected and you are using a database other than Db2.

Bypass upgrade version check
Select this option to skip validation of the IBM Maximo Asset Management version you are upgrading to Maximo Manage. Select this option to continue a failed upgrade that failed during the maxinst or updatedb process.

6. If you want to set Maximo Manage with a language different from English or include other languages in your Maximo Manage deployment, clear the **System managed** checkbox in the Languages section.

   Then, in the **Base** field, select your preferred language to be the base language and in the **Additional** field, order the list of other languages. For more information, see Language support.

   Note: If you are selecting other languages, make sure that you do not select a language in the **Additional** field that was selected in the **Base** field. For example, if you set the base as **EN**, do not select **EN** in the **Additional** field.

7. Configure server bundles for your deployment.
   a) If you want to deploy Maximo Manage with more than one server bundle or with customized configurations for it, under the Server Bundles section, clear the **System managed** option.

      A table with **Name**, **Pod count**, **Type**, and **Additional Properties** is displayed. This table has the **Default**, **User synchronization**, and **Mobile** optional fields available.

When you clear the **System managed** option, a line with the server bundle named as *all* is set with one Pod with the Type *all* and Route subdomain as *all*.

b) Click **Add bundle** to add more server bundles.

c) Select which server bundle to set as respectively the Default, the server bundle to be used for user synchronization, and the server bundles for Mobile.

d) Optional: Change the name, pod count, type, route subdomain name and other customized configurations according to your preference.

e) Click the **View** label under **Additional Properties** column to view the **Route subdomain** value.

The **Additional server bundle properties** page for your selected server bundle is displayed with the **Route subdomain** and **Additional server config** fields available and a list of properties you can define by selecting **Add property +** option in the **Bundle level properties** table. For more information, see Server bundle overview.

8. If you want to include specific customizations through a customization archive, clear the **System managed** checkbox in the Customizations section. In the **File address** field, specify the location of the customization archive and if you must enter credentials to access the file, specify them in the **Credentials (optional)** field. For more information, see the related sections in Customizing the application.

a) In the Customization section of the configuration window, specify the URL for the customization archive file.

The following URL protocols are supported:

- HTTP
- HTTPS
- FTP
- FTPS

To include more customization archive files, click **Add customization archive**.

b) Optional: If you applied password security to the file, in the **Credentials** field, specify the user ID and password in the following format:

```
user=your user name password=your password
```

9. If you do not want the server bundles to start after the database operations of the Maximo Manage deployment are completed, clear the **System managed** checkbox in the **Server mode** section.

Then, set **Mode** to **Off** to prevent access to the Maximo Manage application after deployment. You can restart the server bundle or bundles when you change the configuration to **On** and activate Maximo Manage again.

10. To connect to PVCs under the **Persistent volume claims** section, clear the **System managed** to **Off**, and click **Add PVC**. A table with the following columns is displayed.

Option	Description
PVC name	User-defined name of the persistent volume claim, maximum of 63 characters
Volume name	Leave blank as it is provisioned dynamically.
Size	Amount of storage that is required for this persistent claim, for example, 60G
Mount path	Mount path for the volume within the Maximo Manage pod.

When you configure the PVCs in OpenShift Container Platform cluster on deployments, use the default storage class name `StorageClasses ocs-storagecluster-cephfs` to create a **ReadWriteMany** (rwx) PVC. The storage in the volume that you provisioned are available to all server bundles in the workspace. You can also configure a PVC for specific server bundles in a deployment. If you configure a PVC for a server bundle, the mount path that you specify for the server bundle PVC overrides the path that you specify for the deployment.

Tip: To configure a PVC in OpenShift Container Platform cluster on IBM Cloud platform, use the default `StorageClasses ibmc-file-gold-gid` (instead of `StorageClasses ocs-storagecluster-cephfs`) to create a **ReadWriteMany** PVC.

11. If you select **Asset Configuration Manager** or **Aviation** in the list of components, the Build data interpreter section is displayed. If you want to customize the configuration for the build data interpreter (BDI), clear the **System managed** checkbox. You can then specify a BDI version instead of latest. In the **BDI version** field, click **Add instance+**. You can customize each instance by selecting **View** in the Configuration column. The **BDI Configuration** page is displayed. You can change and save the configuration. Then, you can return to the **BDI configuration** page, select **Reset to Defaults**, and click **Save** to return to the default settings.
12. If you want to use an earlier build for deployment, in the Build section, set **System managed** to **Off**.

    Then, in the **Build tag** field, specify the build tag. Build images are tagged with a timestamp, for example `buildtag: 202011092887843`.
13. To connect to any external systems that Maximo Manage with is integrated with, import the certificate for the system.
14. Optional: You might want to specify the time zone that your database server is configured to use. In the Server time zone section, clear the **System Managed** checkbox. In the **Time Zone** field, select the time zone of your database server.
15. If you are deploying Maximo Health as part of Maximo Manage, you can set following configurations.
    - You can enable asset investment optimization. In the Asset investment optimization section, clear the **System managed** checkbox. Then, select **Asset investment optimization**. When asset investment optimization is enabled, the **Asset investment optimizer** page is available in Maximo Health. Ensure that you deploy and configure Maximo Scheduler Optimization before you enable asset investment optimization.
    - If you have IBM Watson Studio and want to use existing models of Maximo Health from Health and Predict - Utilities, in the IBM Watson Studio section, clear the **System Managed** checkbox and then specify the **Watson Studio Project ID**.
    - If you want the Maximo Health to deploy the out of box data loader configuration files and create the integration server, in the IBM App Connect section, clear the **System Managed** checkbox and then specify the **Dashboard URL**.

Activating Maximo Manage

After the deployment process completes, you activate Maximo Manage so that the application is available for users.

About this task

To activate an application, in Maximo Application Suite, from Maximo® Health catalog page, click **Activate**. If the application deployment is not complete, the **Activate** button is inactive.

After the button is selected, it can take over 24 hours for the activation process to complete. An application might appear in Maximo Application Suite and not be ready to use. Activating the application does not automatically grant users access to the application.

To monitor the application activation, as an administrator, log in to the OpenShift web console and from the OpenShift cluster, in the navigation menu, click **Workloads** > **Pods**.

As the application is activated, a health-runtime-*string* pod is added to the table. The status of this pod is Init:0/1. When the status of the pod changes to Running, the application activation is complete.

You can open the maxinst logs to monitor the activation. To open the logs, complete the following steps:

1. Open the health-runtime-*string* pod.
2. Click **Logs**.

3. Change the container value to maxinst.

After the activation is complete, the application is available from the Maximo Application Suite application navigator and at fixed URLs, and you can begin the Post-activation steps.

Procedure

Customer-managed Migrating

You must upgrade to Maximo Application Suite and then deploy Maximo Manage when you upgrade from Maximo Asset Management. You must perform some key tasks and understand important information after you deploy and activate Maximo Manage.

Related concepts

"Installing Maximo Application Suite" on page 29

Report migration

Report components are automatically upgraded as a part of the database migration process. However, you must back up existing report data from Maximo Asset Management and import the data into the newly installed Maximo Application Suite.

Business Intelligence and Reporting Tools (BIRT)

You can use BIRT reporting in Maximo Application Suite. Maximo Application Suite supports BIRT version 4.8.

- When you migrate from Maximo Asset Management 7.6.1.2 to Maximo Application Suite, BIRT reports are automatically upgraded from version 4.3.1 to version 4.8.
- Because Maximo Asset Management 7.6.1.3 already supports BIRT version 4.8, migrating to Maximo Application Suite does not require an upgrade for BIRT.

To migrate existing report properties and designs, export the report properties and design from Maximo Asset Management and import them in Maximo Application Suite. For more information, see Administering reports.

Cognos® Analytics

You can use Cognos Analytics with Maximo Application Suite. Migrating from Maximo Asset Management 7.6.1 to Maximo Application Suite upgrades Cognos Analytics to version 11.2.4. To migrate Cognos Analytics reports, back up your current data from the Cognos content store and import the data in Cognos Analytics version 11.2.4.

External report integration

Any external report integrations are migrated with the customization archive. For more information, see Migrating customizations using customization archive.

Integrating with external systems

Maximo Manage has an in-built Java Messaging Service (JMS) provider and it is possible to use an external JMS provider as well.

About this task

Maximo Application Suite 8.8 onwards supports an in-built JMS provider that you can use for integration purposes in Maximo Manage.

Procedure

1. Procure, install, and configure the external JMS provider. For example, IBM MQ or Liberty messaging engine.
2. Register the JMS provider queues in Maximo Manage.
3. Associate the registered queues by using the **External Systems** application in Maximo Manage. For more information, see Adding JMS queues to an external system.

   Note: If you are using the Liberty messaging engine, you must register the queues and connection factories in Maximo Manage by using the server.xml file for Liberty. For more information, see Enabling JMS messaging for Liberty.

Adding server bundle properties

Use Maximo Application Suite to add or update server bundle properties.

Procedure

1. In Maximo Application Suite, click the **Administration** icon.

   Starting in Maximo Application Suite 9.1, select the **Suite** > **Administration** page.
2. From the side navigation menu, select **Workspace**.
3. On the **Workspace** page, click the **Manage** tile.
4. On the **Manage workspace details** page, click **Actions** and then click **Update configuration**.
5. In the **Update Manage configuration** dialog, in the **Activation configuration** section, click the **Edit** icon for **Server bundles**.
6. In the **Server bundles** section, click **View** for the server to which you want to add properties.
7. In the **Additional server bundle properties** dialog, in the **Bundle level properties** section, add the bundle-specific properties .
8. Click **Save**.
9. Click **Activate** to activate your changes.
10. Validate the bundle-level properties by using the following API call.

```
{UI_SERVER_URL}/api/service/system?
action=wsmethod:getProperty&propName=mxe.int.webappurl&apikey={API KEY}
```

Sample custom resource:

```
{
    "spec": {
        "bindings": {
            "jdbc": "workspace-application"
        },
        "components": {
            "base": {
                "version": "latest"
            }
        },
        "settings": {
            "db": {
                "maxinst": {
                    "bypassUpgradeVersionCheck": false,
                    "db2Vargraphic": true,
                    "demodata": false,
                    "indexSpace": "maximo",
                    "tableSpace": "maximo"
                },
                "dbSchema": "MAXIMO"
            },
            "deployment": {
                "serverBundles": [
                    {
                        "bundleType": "ui",
                        "isDefault": true,
                        "name": "default",
```

```
                        "replica": 1,
                        "routeSubDomain": "ui",
                        "bundleLevelProperties": "mxe.int.webappurl=http://localhost/
ui\nmxe.webclient.activitydashboard=TRUE"
                    },
                    {
                        "": "cron",
                        "replica": 1,
                        "bundleType": "cron",
                        "isDefault": false,
                        "routeSubDomain": "cron",
                        "bundleLevelProperties": "mxe.int.webappurl=http://localhost/
cron\nmxe.cronTaskMonitorInterval=30"
                    },
                    {
                        "name": "report",
                        "replica": 1,
                        "bundleType": "report",
                        "isDefault": false,
                        "routeSubDomain": "report",
                        "bundleLevelProperties":
"mxe.report.reportsInAPage=10\nmxe.int.webappurl=http://localhost/report"
                    },
                    {
                        "name": "mea",
                        "replica": 1,
                        "bundleType": "mea",
                        "isDefault": false,
                        "routeSubDomain": "mea",
                        "bundleLevelProperties":
"mxe.adminEmail=email@ibm.com\nmxe.int.webappurl=http://localhost/meaweb"
                    }
                ]
            }
        }
    }
}
```

Server bundles

You can add or update server bundle properties when you deploy IBM Maximo Manage. For each server bundle, a service and a route that points to the service are created.

When the IBM Maximo Manage application is deployed in IBM Maximo Application Suite, a Maximo Manage deployment custom resource (CR) is created.

- The Maximo Manage deployment CR contains the configuration that you entered, which includes database URL, server bundle types, deployment size, and other information.
- The database username or password can be specified in Maximo Application Suite and persisted as a secret in the cluster.
- The Maximo Manage **crypto** or **cryptox** properties can be updated in Maximo Application Suite and persisted as a secret in the cluster.
- Both secrets, the username or password and the **crypt** or **cryptox** values can also be updated in Red Hat OpenShift. Any change to the values automatically redeploys the application completely or partially.

Server bundle

A server bundle or workload is a logical abstraction for a deployed group of point of deployment (pods) in a cluster. These server bundles do the same function and provide an access point as a service. These server bundles can be accessed as a service internally and through a route externally. A route is a way to display a service by giving it an externally reachable hostname. By using a route or service, Red Hat OpenShift balances the load to the pods that are included in a server bundle. Each server bundle defines replica size, subdomain, and other properties.

- For each server bundle, a service is created with name that is appended by **-&ltserverbundlename&gt**.
- A route is created with name that is appended by **-&ltserverbundlename&gt**.
- A default route is created to point to the service that ends with **-&ltserverbundlename&gt**.

  - Maximo Application Suite uses the default route to establish the default URL link to Maximo Manage.
- You can add or update server bundle properties.

The following diagram illustrates how Red Hat OpenShift Container Platform routers provide external hostname mapping and balancing of load for service end points over protocols. The router uses the hostname to determine where to send the external client request.

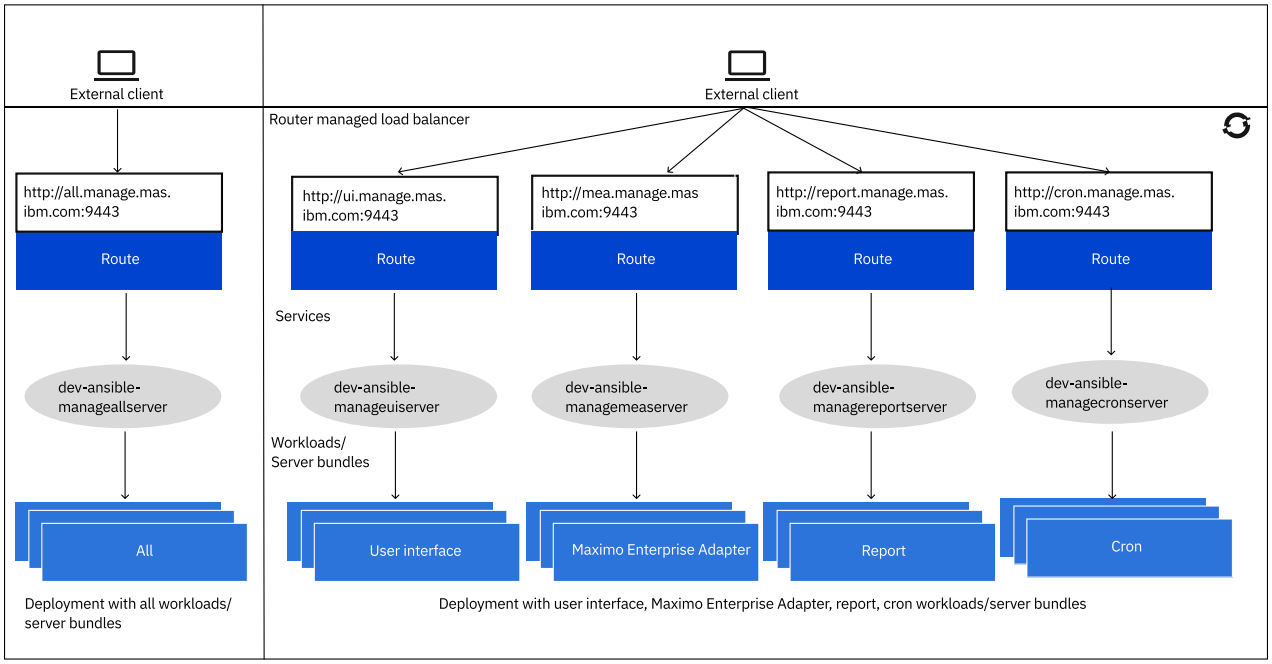


Figure 5. External host mapping and load balancing

- The Maximo Manage application can be deployed with different server bundles or workloads for the processing and isolation needs.
- The deployment can be All bundle server type or a combination of the four bundle server types: UI, cron, report, and Maximo Enterprise Adapter.
  - If All bundle server type is not deployed, and you used a combination of the four bundle server types, you must use the UI bundle server type.
- Each server bundle can have its own server properties.

The following table shows the five different server bundle types.

Bundle server type	**Description**
All	Contains all the code.
UI	Contains UI code and supporting code. It is the interface for accessing Maximo Manage application.
Maximo Enterprise Adapter	Displays the enterprise web services API.
report	Contains the code that is needed to enable BIRT Report Only Server (BROS). It is used to separate the work load that is related to the execution of reports that are submitted in Maximo Manage.
cron	This bundle contains the code that is needed to run Maximo Manage cron tasks.

Server bundle properties

The server bundles have the following properties:

- The server bundle properties can be set in Maximo Application Suite UI or in the CR.
- A configmap `<workspaceid>-<serverbundlename>-bundleproperty` is created for the server properties during deployment or operator reconciliation. It is mounted to `/config/manage/properties` on the pod. The **`bundleLevelProperties`** file on the pod must not be updated manually.
- Maximo Manage server process automatically detects the change and updates the Maximo Manage property cache.
- If the property value needs to be modified, update the CR directly or in Maximo Application Suite. The Maximo Manage Operator reconciles the changes. The Maximo Manage server process updates the Maximo Manage property cache with the updated value.

Liberty server XML

You might need to customize the Liberty `server.xml` file, for example, when creating queues.

- The custom server XML can be set in Maximo Application Suite UI. For example, to create queues in JMS server.
  - If you create it in the CR directly, then you must create the secret manually. Set this secret to **`additionalServerConfig.secretname`** for the bundle server in the CR.
- On deployment, a secret is created for the custom `server.XML` file. It is mounted to **`/config/manage/serverxml`**. This location is included in Liberty `server.xml` file. The configuration is applied to the Liberty server.

Sample custom `server.xml` file:

```
<?xml version="1.0" encoding="UTF-8"?>
<server description="new server">
    <featureManager>
          <feature>wmqJmsClient-2.0</feature>
     <feature>jmsMdb-3.2</feature>
    </featureManager>
    <logging traceSpecification="JMSApi=all:WAS.j2c=all"/>
    <variable name="wmqJmsClient.rar.location" value="/wmq/wmq.jmsra.rar"/>
    <jmsConnectionFactory jndiName="jms/maximo/int/cf/intcf" connectionManagerRef="MIFJMS">
         <properties.wmqJms
            transportType="CLIENT"
            hostName="mifjmsmanager-afd7.qm.us-south.mq.appdomain.cloud"
            port="31440"
            channel="CLOUD.APP.SVRCONN"
            applicationName="maxliberty"
            userName="{{username}}
            password="{{yourpasssword}}"
            queueManager="MIFJMSMANAGER"/>
    </jmsConnectionFactory>
    <connectionManager id="MIFJMS" maxPoolSize="20"/>
    <jmsQueue id="sqout" jndiName="jms/maximo/int/queues/sqout">
      <properties.wmqJms baseQueueName="sqout" baseQueueManagerName="MIFJMSMANAGER"/>
    </jmsQueue>
    <jmsQueue id="sqin" jndiName="jms/maximo/int/queues/sqin">
      <properties.wmqJms baseQueueName="sqin" baseQueueManagerName="MIFJMSMANAGER"/>
    </jmsQueue>
    <jmsQueue id="jms/maximo/int/queues/cqin" jndiName="jms/maximo/int/queues/cqin">
      <properties.wmqJms baseQueueName="cqin" baseQueueManagerName="MIFJMSMANAGER"/>
    </jmsQueue>
    <jmsQueue id="jms/maximo/int/queues/cqinerr" jndiName="jms/maximo/int/queues/cqinerr">
      <properties.wmqJms baseQueueName="cqinerr" baseQueueManagerName="MIFJMSMANAGER"/>
    </jmsQueue>
</server>
```

User migration

You can create new users in IBM Maximo Application Suite users or migrate from Maximo Asset Management.

Migrated users

Users are migrated to Maximo Application Suite from IBM Maximo Asset Management during the upgrade.

Users are created with the **Set in Manage** access type. The user access is managed in the IBM Maximo Manage application by using security groups. For more information, see Configuring security groups.

If SMTP is configured, the migrated users receive the following emails:

- Welcome to IBM Maximo Application Suite.
- Your IBM Maximo Application Suite password.

On the **Users** page, you can replace passwords in edit mode by using the **Replace forgotten password** link. A MAXADMIN user is created by default. User authorization or application access is done in the Maximo Manage application by using security groups. .

Customer-managed Mapping LDAP fields as person ID for Maximo Manage in Maximo Application Suite 8.11

In Maximo Application Suite 8.11 and earlier versions, the person ID is equal to the user ID in Maximo Manage. If you are migrating users from Maximo Asset Management to Maximo Application Suite from an LDAP server and the person ID is different than the user ID, you can configure the user data to maintain the person ID data.

Before you begin

Before you can configure the user data to maintain the person ID, install Maximo Application Suite 8.11.7 or 8.10.10 fix packs or later. You can install fix packs by using a channel subscription. For more information, see Upgrading IBM Maximo Application Suite by using the channel subscription method.

If you are using Maximo Application Suite 9.0, see “Mapping LDAP fields as person ID for Maximo Manage in Maximo Application Suite 9.0” on page 114

About this task

When a user is added in Maximo Application Suite, the user ID is created as the primary identifier for that user. When user synchronization occurs with Maximo Manage, the user ID is added in Maximo Manage as the user ID and the person ID.

If you need to maintain separate person ID data to meet your business needs, create an automation script in Maximo Manage that copies the value from the LDAP `employeeNumber` field to the `personid` field during the user synchronization process. You can also use this automation script to map to other fields in Maximo Manage.

If the field you are using in the LDAP server is not `employeeNumber`, you must also map the field to `employeeNumber` in the `ScimCfg` LDAP configuration record.

Procedure

1. Create an automation script that copies the value from the `employeeNumber` field to the `personid` field,
   a) In Maximo Manage, open the Automation Scripts application.
   b) From the **More Actions** menu, select **Create** > **Script for Integration**.
   c) In the Integration Details section, select **Enterprise Services** and in the Enterprise Service field, select MASPERUSER.
   d) Select **Request**, **User Exit**, and **After External Exit**.
   e) In the Script Details section, enter the following code as Jython code:

```
from com.ibm.tivoli.maximo.oslc import OslcUtils
from com.ibm.tivoli.maximo.oslc.provider import OslcJSONStructureData
```

```
data = irData.getDataAsBytes()
dataEr = erData.getDataAsBytes()

jo = OslcUtils.bytesToJSONObject(data)
erJo = OslcUtils.bytesToJSONObject(dataEr)

if erJo.get("owner").upper() == 'SCIM':
   extjo = jo.get("extension")
   empno = extjo.get("employeeNumber")
   jo.put("personid",empno)
   irData = OslcJSONStructureData(jo,"MASPERUSER","PERSON",userInfo,"Sync",True)
```

2. If the field in the LDAP server is not `employeeNumber`, update the `ScimCfg` LDAP configuration record to map the field to `employeeNumber`.
   a) In Red Hat OpenShift, from the side navigation menu, click **Administration** > **CustomResourceDefinitions**.
   b) On the **CustomResourceDefinitions** page, search for and open the `ScimCFG` custom resource definition.
   c) On the **Instances** tab, search for and open the CR that starts with the Maximo Application Suite instance ID.

      For example, *<your_mas_instance>*`-scim-default-system`.
   d) On the **YAML** tab, in the `spec:` section under `usersync`, map the field to `employeeNumber`.

   For example, if the field in LDAP is named `employeeID`, add the following configuration in `spec.usersync` to map to `employeeNumber`.

```
userSync:
  mappings:
    extensions:
      employeeNumber: employeeID
```

Customer-managed

Mapping LDAP fields as person ID for Maximo Manage in Maximo Application Suite 9.0

In Maximo Application Suite 9.0, the person ID is equivalent to the user ID in Maximo Manage. If you are migrating users from Maximo Asset Management to Maximo Application Suite from an LDAP server and the person ID is different than the user ID, you can configure the user data to maintain the person ID data.

Before you begin

If you are using Maximo Application Suite 8.11 or earlier, see “Mapping LDAP fields as person ID for Maximo Manage in Maximo Application Suite 8.11” on page 113

About this task

When a user is added in Maximo Application Suite, the user ID is created as the primary identifier for that user. When user synchronization occurs with Maximo Manage, the user ID is added in Maximo Manage as the user ID and the person ID.

If you need to maintain separate person ID data, you can customize the mapping for the person ID to synchronize with the value from the LDAP server during the user synchronization process. For example, you can customize the mapping for the person ID to map to the `employeeNumber` field in LDAP.

Procedure

1. On the **Suite administration** page, from the side navigation menu, select **Configurations**, click **User registry synchronization** and then **Edit**.
2. Specify the following LDAP domain attributes for the LDAP server.

LDAP URL
The URL for the LDAP server.

Base DN
The path in the object hierarchy of the directory server.

Bind DN
The user and location that is used to bind to an LDAP server.

Maximum user synchronization

The maximum number of users that are synchronized between the LDAP server and the user registry. If the search results of the LDAP database exceed this limit, the synchronization process is canceled. This property is the `customMaxSearchResults` property in the `ScimCfg` custom resource.

LDAP type
The type of LDAP server that you are using. For example, select Microsoft Active Directory if that is the LDAP server that you are using.

3. Add or retrieve the CA certificate for the LDAP server.
4. Specify the user synchronization, such as User Base DN, ID map, and filter.
5. In the User mapping section, switch **Use default mapping** to off and enter the custom mapping for the person ID user data to synchronize with the employee number from the LDAP server.
   a) Click **Add custom mapping**.
   b) In the Maximo Application Suite field column, enter `person.personid` in the field after `extension`.
   By specifying the `person` value, the `personid` maps to the person table in Maximo Manage
   c) In the LDAP field column, enter `employeeNumber`.
6. Optional: Specify group synchronization information, such as Group Base DN, filter, ID map, and member ID map.
7. Optional: Enter the custom mapping for the group data to synchronize with the LDAP server.
8. Optional: Assign the default application entitlement and application access to apply to all synced users.
   You can also modify the entitlement and access for individual synced users on the **Users** page.

Results

When you save the mapping changes, the configuration is processed, and the user synchronization changes are applied in the next scheduled synchronization.

Managing users post upgrade

Migrated users are synchronized by a cron task, and you can view the synchronization information. If your migrated users have different user IDs and login IDs and you are upgrading to Maximo Application Suite 8.7 or earlier, you must re-create your existing users. User entitlement to applications and AppPoints, and license consumption reports are viewable in the **Suite administration** dashboard. You can also change user passwords by editing the user profile on the **Users** page in Maximo Application Suite.

Synchronizing migrated users

The MASUSERSYNC cron task synchronizes users from Maximo Manage to Maximo Application Suite.

Before you begin

You must have administrator rights to set up the MASUSERSYNC cron task instance. If the users are not migrated to Maximo Application Suite, check the database log and fix any errors.

Procedure

1. On the **Users** page in Maximo Application Suite, create an administrative user . For more information, see Administering users and user access.
2. Log in to Maximo Application Suite by using the administrative user's credentials.
3. Click the **Administration** icon to open the **Suite navigator** page and then click the Manage tile.
4. From the side navigation menu, click **System Configuration** > **Platform Configuration** > **Cron Task Setup**.

Note: Starting in Maximo Application Suite 9.1, the Manage application is available in the side navigation menu. You can access the **Cron Task Setup** page directly from the side navigation menu in **System configuration** > **Platform configuration** under the Manage application.

5. Search for and select the MASUSERSYNC cron task.
6. On the **Cron Task** tab, click **New Row**.
7. Specify a name and a schedule for the cron task instance.

   The date is shown as a string in the **Schedule** field. Do not change the string in the **Schedule** field. Click **Set Schedule** to change the schedule.
8. Specify a user who has the necessary privileges. The user must have access for the actions that the cron task performs.
9. Optional: Select the **Active** check box if you want to activate the cron task.
10. Save your changes.
11. Select the **Reload Request** action.
12. Select the instance and click **OK** to run the cron task.

User synchronization information

You can view information about user synchronization in IBM Maximo Manage.

- When a user is created and is assigned the Manage role, the user record is synchronized into Maximo Manage. The user sync process pulls the data from the user registry in IBM Maximo Application Suite and pushes it to Maximo Manage.
- If a user is assigned to the NO_ACCESS role or is deleted in Maximo Application Suite, the user becomes inactive in Maximo Manage.
- The user synchronization process from Maximo Application Suite to Maximo Manage uses the Maximo integration framework. It uses an enterprise service to post data to Maximo Manage. If a IBM Maximo Application Suite user is assigned the Manage role and the user record has a sync status of PENDING or DELETE_PENDING, the user synchronization process processes the user record. After successful synchronization, the status is changed to SYNCED.

Sample of inbound message from Maximo Application Suite


```
{ "_id": "joesmith5", "username": "joesmith5", "permissions": { "systemAdmin": true,
"userAdmin": true }, "issuer": "local", "displayName": "Joe Smith", "familyName": "Smith",
"givenName": "Joe", "title": "Supervisor", "preferredLanguage": "EN", "locale": "en_US",
"phoneNumbers": [ { "value": "555-555-5555", "type": "work" }, { "value": "555-555-4444",
"type": "work" } ], "addresses": [ { "streetAddress": "100 Universal City Plaza", "locality":
"Hollywood", "region": "CA", "postalCode": "91608", "country": "USA", "formatted": "100
Universal City Plaza\nHollywood, CA 91608 USA", "type": "work", "primary": true } ], "emails":
[ { "value": "joesmith5@us.ibm.com", "type": "work", "primary": true }, { "value":
"joesmith51@gmail.com", "type": "work" } ], "extension": { "employeeNumber": "701984",
"costCenter": "4130" }, "entitlement": { "application": "PREMIUM", "admin": "ADMIN_PREMIUM" },
"workspaces": { "space1": { "permissions": { "workspaceAdmin": true }, "applications":
{ "manage": { "role": "ADMIN" }, "iot": { "role": "ADMIN" }, "health": { "role": "USER" },
"monitor": { "role": "ADMIN" } } } }, "added": { "id": "admin", "timestamp":
"2020-08-10T18:01:36.694331" }, "updated": { "id": "admin", "timestamp":
"2020-08-10T18:08:17.455782" }, "sync": { "status": "SUCCESS", "timestamp":
"2020-08-10T18:10:16.731047" }, "applications": { "manage": { "sync": { "state": "PENDING",
"reason": "", "timestamp": "2020-08-10T18:08:49.604430" } }, "monitor": { "sync": { "state":
"SUCCESS", "reason": "", "timestamp": "2020-08-10T18:08:17.471947" } }, "health": { "sync":
{ "state": "SUCCESS", "reason": "", "timestamp": "2020-08-10T18:09:18.131171" } }, "predict":
{ "sync": { "state": "SUCCESS", "reason": "", "timestamp": "2020-08-10T18:08:17.471947" } } },
```

```
"token":
"1000:8744077b1411c0601e4912d556d93ff859089bfd16863f16:591949ad7b4d4f7017de846a3f3b2609ac4caef4e
be09448" }
```

Sample of output from default user exit

```
{ "addressline1": "100 Universal City Plaza", "city": "Hollywood", "country": "USA",
"displayname": "Joe Smith", "email": [ { "_action": "AddChange", "emailaddress":
"joesmith5@us.ibm.com", "isprimary": 1, "type": "work" }, { "_action": "AddChange",
"emailaddress": "joesmith51@gmail.com", "isprimary": 0, "type": "work" } ], "extension":
{ "costCenter": "4130", "employeeNumber": "701984" }, "firstname": "Joe", "language":
"EN", "lastname": "Smith", "locale": "en_US", "maxuser": [ { "groupuser": [ { "_action":
"AddChange", "groupname": "MAXADMIN" }, { "_action": "AddChange", "groupname": "TOOLMGR" } ],
"inactivesites": 1, "loginid": "joesmith5", "statusdate": "2020-10-30T15:43:44-04:00",
"userid": "joesmith5" } ], "personid": "joesmith5", "phone": [ { "_action": "AddChange",
"isprimary": 1, "phonenum": "555-555-5555", "type": "work" }, { "_action": "AddChange",
"isprimary": 0, "phonenum": "555-555-4444", "type": "home" } ], "postalcode": "91608",
"stateprovince": "CA", "statusdate": "2020-10-30T15:43:44-04:00", "title": "Supervisor" }
```

Troubleshooting user migration

The migration process uses an internal API to migrate user IDs from Maximo Asset Management to Maximo Application Suite. If you have users with a different user ID and login ID and you are upgrading to Maximo Application Suite 8.7 or earlier, you must re-create your existing users.

About this task

Maximo Application Suite has two fields for user management, username and user ID. If any of your Maximo Asset Management users have a user ID different from their login ID, you must delete and re-create the users after you upgrade to Maximo Application Suite.

Procedure

1. If the user has only Maximo Manage license entitlement:
   a) In the Maximo Application Suite user interface, login as an administrator and go to **Suite Administration** > **Users**.
   b) Search for the user by using the Maximo Asset Management login ID as the search criterion.
   c) Create a new user based on the information of the existing user and set the user ID as a unique key and login ID.

      For more information, see Administering users and user access.
   d) Delete the old user.
   e) Run the **MASUSERSYNC** cron task to synchronize the users from Maximo Application Suite to Maximo Manage

      For more information, see Synchronizing migrated users.
2. If the user has entitlement to multiple products in Maximo Application Suite, for example, Maximo Manage and IoT:
   a) In the Maximo Application Suite user interface, login as an administrator and go to **Suite Administration** > **Users**.
   b) Create a user based on the information of the existing user and set the user ID as a unique key and login ID.
   c) Change the login ID of the existing user by adding **_bk** to it.
   d) Delete the email address of the existing user to avoid duplicate email addresses.
   e) Do not synchronize the user to any products in Maximo Application Suite.

      Tip: In the case of a user in multiple products, preserve the original user information from Maximo Asset Management even if the user cannot log in to the system by using the Maximo Asset Management login ID. If you want to keep an old user, you can change the **Username** and use

the user. If you decide to delete it, you can delete it yourself and add the entitlement to the new user. To find these users easily, search for **Usernames** with "_bk".

User entitlement

License entitlement for Maximo Manage can be determined by using a cron task, `MasUserAnalyzer`.

- In Maximo Manage, use a cron task to calculate user entitlement.
  - The `MasUserAnalyzer` cron task calculates the user entitlement, which can be Limited, Base, Premium, or Self-Service.
  - The `MasUserReporting` cron task sends the entitlement change to Maximo Application Suite.

    Note: If the entitlement is greater in Maximo Manage, a message is displayed on the **Suite administration** dashboard.
  - For more information on how the cron task works, see License type information.

To generate License consumption reports, open the **Suite administration** dashboard and click **License consumption**.

Changing user passwords

User passwords are changed on the **Users** page in Maximo Application Suite.

Procedure

1. Maximo Application Suite, from the side navigation menu, select **Users**.
2. Click on the user record to go to **View user** page.
3. Click the **Edit** icon to edit the user details.
4. In the **Authentication** section, click **Replace forgotten password**.
5. Select **Send password by email** to receive an email that contains the password.
6. Select **Autogenerated** to automatically generate the password or select **Custom** to manually specify the new password.
7. Click **Save changes**.

Changing current user password

On the**Suite administration** dashboard, you can change the password for the currently logged in user.

Procedure

1. In Maximo Application Suite, in the main menu bar, click the **Profile** icon.
2. Click **Manage profile**.
3. On the **Change password** tab, specify the current password and the new password and then click **Save**.

Updating system settings path

Update system settings with the new values.

About this task

The system settings depend on your current Maximo Application Suite setup. For example,

- `mxe.doclink.path01`
- `mxe.doclink.doctypes.defpath`
- `mxe.doclink.doctypes.topLevelPaths`

Secured attachments must always be enabled for Maximo Application Suite or Maximo Manage.

Procedure

1. Update the path in the doc-link table:

```
update docinfo set urlname = replace(urlname, 'C:','');
```

```
update docinfo set urlname = replace(urlname, '\','/');
```

```
update docinfo set urlname = replace(urlname, 'DOCLINKS','doclinks');
```

2. Verify the doc information table because the directories are case-sensitive in Maximo Manage.

Using certificates

You can use route or TLS certificates, SSL certificates, external certificates.

Route or TLS certificates

Maximo Application Suite uses cert-manager for automatic management and issuance of TLS certificates for application routes. During installation, you can provide a cluster issuer that is based on a trusted certificate authority (CA) for signing the certificates that are generated for your Maximo Application Suite domains.

By default, Maximo Application Suite provides a cluster issuer that generates Maximo Application Suite certificates that are signed by a self-signed CA. To use your own cluster issuer, include the following parameter when you run the Maximo Application Suite installer:`-c myClusterIssuerName`. For more information, see System requirements.

Obtaining SSL certificate for database

If the database requires an SSL connection, you must obtain the certificate for the database. You can use the **openssl** command to connect to the database host and port that is specified in the database URL

Procedure

1. Run the **openssl** command: `openssl s_client -showcerts -connect databasehost:databaseport`
2. While activating Maximo Manage, go to the **Database** section and open it.
3. Give the certificate an alias that does not conflict with other certificates in the trust store.
4. Make sure that **SSL Enabled** is set to Yes in the UI.
5. Specify the certificate on the jdbccfg CR in the certificates section as shown in the following example.

   Sample Custom Resource (CR)

```
apiVersion: config.mas.ibm.com/v1
kind: JdbcCfg
metadata:
  name: "mng-jdbc-system"
  namespace: "mas-mng-core"
  labels:
    mas.ibm.com/configScope: system
    mas.ibm.com/instanceId: "mng"
spec:
  displayName: IBM Cloud Databases for Db2
  config:
    url: "jdbc:db2://dashdb-txn-sbox-yp-lon02-02.services.eu-gb.bluemix.net:50001/
BLUDB;sslConnection=true"
    sslEnabled: true
    credentials:
      secretName: db2-masdev-lite-credentials
  certificates:
```

```
  - alias: part1
    crt: |
      -----BEGIN CERTIFICATE-----
      MIIG7zCCBdegAwIBAgIQBMX5yCODP3RIcSrij07HJjANBgkqhkiG9w0BAQsFADBN
      MQswCQYDVQQGEwJVUzEVMBMGA1UEChMMRGlnaUNlcnQgSW5jMScwJQYDVQQDEx5E
      aWdpQ2VydCBTSEEyIFNlY3VyZSBTZXJ2ZXIgQ0EwHhcNMjAwMjA3MDAwMDAwWhcN
      MjIwMjExMTIwMDAwWjCBkzELMAkGA1UEBhMCVVMxDzANBgNVBAgTBkthbnNhczEW
      MBQGA1UEBxMNT3ZlcmxhbmQgUGFyazE0MDIGA1UEChMrSU5URVJOQVRJT05BTCBC
      VVNJTkVTUyBNQUNISU5FUyBDT1JQT1JBVElPTjElMCMGA1UEAwwcKi5zZXJ2aWNl
      cy5ldS1nYi5ibHVlbWl4Lm5ldDCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoC
      ggEBAJzzmjnE+tJklZkrEEHP4RkDdJ+Tmho0diJdK8soPNVSvs5SuWNWKnbDDohE
      B/jEiSGmClWKPFzAAn+PlTU2djntLma6LWATycOzx0E2r+22la1yg+UhZrOBfQ9e
      fAy3wQcu7Aylsq8OsMgM33+U0aM254urrkg3x0RV1Do1Y1sAzw2/wJmSilmMOvyf
      hWZUad9hNWi/1bEt6z4WPn/23lbvDzeTlr6jznHFArQ8e6/AV98orFd82NxZcM6K
      ByGtQpyXkBNf0wy/8kbyY0qzisoWLnanyxAHJABnUEsjS3WmvRQ8H1QW7RT63lwL
      9s13dErNg7D8l4L9NVWsiJMaTPcCAwEAAaOCA4IwggN+MB8GA1UdIwQYMBaAFA+A
      YRyCMWHVLyjnjUY4tCzhxtniMB0GA1UdDgQWBBSuP2nUw2cmgLQqvH6vI7Ok1zQ/
      jjBDBgNVHREEPDA6ghwqLnNlcnZpY2VzLmV1LWdiLmJsdWVtaXgubmV0ghpzZXJ2
      aWNlcy5ldS1nYi5ibHVlbWl4Lm5ldDAOBgNVHQ8BAf8EBAMCBaAwHQYDVR0lBBYw
      FAYIKwYBBQUHAwEGCCsGAQUFBwMCMGsGA1UdHwRkMGIwL6AtoCuGKWh0dHA6Ly9j
      cmwzLmRpZ2ljZXJ0LmNvbS9zc2NhLXNoYTItZzYuY3JsMC+gLaArhilodHRwOi8v
      Y3JsNC5kaWdpY2VydC5jb20vc3NjYS1zaGEyLWc2LmNybDBMBgNVHSAERTBDMDcG
      CWCGSAGG/WwBATAqMCgGCCsGAQUFBwIBFhxodHRwczovL3d3dy5kaWdpY2VydC5j
      b20vQ1BTMAgGBmeBDAECAjB8BggrBgEFBQcBAQRwMG4wJAYIKwYBBQUHMAGGGGh0
      dHA6Ly9vY3NwLmRpZ2ljZXJ0LmNvbTBGBggrBgEFBQcwAoY6aHR0cDovL2NhY2Vy
      dHMuZGlnaWNlcnQuY29tL0RpZ2lDZXJ0U0hBMlNlY3VyZVNlcnZlckNBLmNydDAM
      BgNVHRMBAf8EAjAAMIIBfwYKKwYBBAHWeQIEAgSCAW8EggFrAWkAdwCkuQmQtBhY
      FIe7E6LMZ3AKPDWYBPkb37jjd80OyA3cEAAAAXAh/gncAAAEAwBIMEYCIQC75oq7
      nsysXvTj9uwOH+4p3/LZD4mEGzC27BQPaFFsvwIhAKinjPXsguG3Q/7EZJey/orx
      hTjJDio6yKKvGQr8SIysAHYAIkVFB1lVJFaWP6Ev8fdthuAjJmOtwEt/XcaDXG7i
      DwIAAAFwIf4KMgAABAMARzBFAiEAztl/4xYHslgBIauTztvYEkMxVFijFKvYAXF3
      v8FzDzMCIA59GDHJfiqC18angaQzKyDIuYuRPTDMNixgoOht/mjUAHYAQcjKsd8i
      RkoQxqE6CUKHXk4xixsD6+tLx2jwkGKWBvYAAAFwIf4JrgAABAMARzBFAiEAsB5k
      q51RvxRXk0wyTWX92QQc+IrL5+0rUwfBXeY1jPgCIH944k+IpKAFN5vM5YnGk/Xl
      ryONIoMWkDOtqdpvyqARMA0GCSqGSIb3DQEBCwUAA4IBAQBShftEwr1tIjh4alof
      Sc+BScv7NdRXHIOJDg2lQZ3mhBq7MttW/cAWpVlEKvEw/31KB2iQLJN90Q8grTwr
      NMBSeNu4b1CTJY+vBYRKKfYEqJH74oHURu4d+9wZI0ZUcHJvXj1vgBR/8O+7YV2Y
      yO2u/4sJJjp3yVNa/RzroI6oS+O1w0znzc5Io+vst50hveVmiwaHH4fNUX00BqHE
      Asy2nFSpvzNS/dlMGgM6XoEU46CMS0ORIoxoMEWRbDk2OPdCtKsg+ySkIYS/ylyN
      vdClL1WOhHqLrG5ZCVQoVgr92vLtxys+rHAeqJISdq3ol6QV3iGpBXjv9hwW9hpi
      XhSQ
      -----END CERTIFICATE----
```

Importing certificates in Maximo Application Suite

You can import certificates into Maximo Application Suite. For example, Maximo Manage might integrate with external systems that contain self-signed certificates, such as Kafka or an SMTP server. You must give an alias to the certificate so that this name is used as the alias for this certificate when you add it to the truststore of Liberty.

About this task

The actual certificate must contain the `-----BEGIN-----` and `-----END CERTIFICATE-----` lines when you copy and paste the certificate itself. It must be in the PEM format. In a custom resource, the certificate content must be added in the `crt` property, and because it is multiline, you must add `|`. You can specify multiple certificates that each have a different alias.

Procedure

1. Log in to Maximo Application Suite as a system administrator.
2. On the side navigation menu, in **Suite** > **Administration** page, click **Configurations**.
3. In the **Other** section, click the row where the external system that Maximo Manage is integrating with is displayed, for example, Apache Kafka.
4. In the dialog, click the Edit icon.
5. Click **Add** to add the certificate.
6. Specify a certificate alias and paste the certificate content in PEM format.
7. Click **Confirm**.
8. Click **Save**.

The following example shows a custom resource (CR).

```
deployment:
    ...
    importedCerts:
        - alias: kafka
          crt: |
            -----BEGIN CERTIFICATE-----
            MIIDLTCCAhWgAwIBAgIJAIyuocAUfASaMA0GCSqGSIb3DQEBCwUAMC0xEzARBgNV
            BAoMCmlvLnN0cmltemkxFjAUBgNVBAMMDWNsdXN0ZXItY2EgdjAwHhcNMjEwNDAx
            MjIzMjUyWhcNMjIwNDAxMjIzMjUyWjAtMRMwEQYDVQQKDAppby5zdHJpbXppMRYw
            FAYDVQQDDA1jbHVzdGVyLWNhIHYwMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
            CgKCAQEAsGgoHx3zN5qJsO6L/fOr7az9eY3sHH1Ne9cZpe8awMtTQD35thL9sT8g
            ahStB0uhE/KbhFuZGcKT1Q8w3vJxaqSLeepTgfk9YWMF0lZbrOXbEg8v+apuvlLN
            a91EI46yuizcKUMLmA7WwqFlCHAaca68z5nkdPDf2BvB2Tmy1UkayjiDm9sPukUE
            qRxdWTw7ZOj8PSBt2KZP9xxyCmA6F7M7KuPr7OOwH+0291mAvzMmp1f/1bg2jw9e
            rz38jmcXrXVVx6I2otJHjTY+wRGEVWafP5vwEt4vNPXHtvi+e3w+HXAvGEstUJQo
            tzPf66+sFfXUI4sT80jJbPfMxrfPgwIDAQABo1AwTjAdBgNVHQ4EFgQU63t8rSXJ
            sJMojAY0wNEHJ/CkjaYwHwYDVR0jBBgwFoAU63t8rSXJsJMojAY0wNEHJ/CkjaYw
            DAYDVR0TBAUwAwEB/zANBgkqhkiG9w0BAQsFAAOCAQEAFKEVmjYplFbM9pfYHpwk
            5+OlNnD3JaCCXDdpP3ccH5ABpxLsD0jf/cl2sUxEoxe+h1LnsxUBVnr6QuU4IIpe
            5OBT82SNZoJUl9w9Qp5IYeb2KF4oCbb80bejz6RXdJsuMk4pxKo1E1tIDYmuXZi7
            Wk8Np588ZHUdzka7dZklr9CtDLywuJGrHxc0t8R2wcqFvGAANG8vMMzuu3DTWk+d
            eMumY6m0Q/BnvPrIrRL1/45GvOv23G5oDLGLkSNmM3UIH+6q18z/vyN5D9K07xUn
            ThpsmIQVwXOiO79qcZxBAceMKlCTxwdMUTg2MfImTqc66/SAj/cIZEN+32OgxEnI
            ZA==
            -----END CERTIFICATE-----

        - alias: smtp
          crt: |
          -----BEGIN CERTIFICATE-----
            ...
          -----END CERTIFICATE-----
```

Verifying the migration

After you have migrated to Maximo Manage, you must check some aspects so that it works as it is supposed to.

Running Integrity Checker after upgrade

The Integrity Checker is a database configuration utility that you can use to check the health of the database that you migrated from Maximo Asset Management to Maximo Manage. To run Integrity Checker, use the `ToolsAPI` API.

Before you begin

To work with APIs, you must use a browser or an API platform, such as Postman.

About this task

Authorized users must have an API key and must belong to the security group that enables the signature options for the APIs for script commands. The object structure that specifies the necessary signature options is available to the default administrator group, such as MAXADMIN. The default administrator group is specified by the value of the *ADMINGROUP* variable name in the MAXVARS table.

Procedure

1. Log in as an administrator in Maximo Manage and in the API Keys application, create API keys for the users. This action registers the keys to the users. Copy the keys and provide them to your users. An API key is displayed only one time and cannot be retrieved later.
2. In the Security Groups application, select the administrator group that can work with the APIs.
3. Select the **Object Structures** tab and search for TOOLSAPI in the **Object Structures** section.
4. From the **Actions** menu, click **Grant Listed Object Structures** and select an access level from the list.

5. In the **Options for Tool APIs** section, select the row for the signature option and click **Grant Listed Options for This Object Structure** to grant access to the signature options that are associated with the APIs.

   Important: Stop the Maximo Manage pods before you run the Integrity Checker utility.

   For more information on Integrity checker warning and error messages, see Integrity checker messages.
6. **Tip:** You can use the Swagger interface to authorize APIs. For more information see, Swagger Interface for APIs.

   Use any of the APIs that are applicable to your upgrade.

API	Action	HTTP method	Signature option	API request type
`icheckerreport`	Generate the integrity checker log.	POST	ICREPORT	Asynchronous
`toolslog`	Get a specified tools log or get a list of all tools logs.	GET	GETLOG	Synchronous
`submitUploadLogRequest`	Upload logs from Maximo Manage UI, Cron, Maximo Enterprise Adapter, or Report pods to S3 Cloud Object Storage.	POST	GETLOG	Asynchronous
`icheckerrepair`	Start integrity checker repair.	POST	ICREPAIR	Asynchronous
`managestart`	Start all Maximo Manage pods.	POST	MANAGESTART	Asynchronous
`managestop`	Stop all Maximo Manage pods.	POST	MANAGESTOP	Asynchronous
`installexternalcert`	Import an external certificate to a server bundle truststore.	POST	INSTALLEXTERNALCERT	Synchronous
`validatedbformas`	Validate the database for the migration to Maximo Manage.	POST	VALIDATEDBFORMAS	Synchronous

   Note: Do not use the **SetAdminMode** command from Manage tools to set the admin mode. The **SetAdminMode** command is not available in Maximo Application Suite. You must use the Maximo Application Suite user interface or use the REST APIs to manage the admin mode.
7. Use any of the following commands in Postman to check the health of your database.

Action	API request
Generate the integrity checker log.	`POST https://`*`host:port`*`/toolsapi/toolservice/icheckerreport`
Get an integrity checker log.	`GET https://`*`host:port`*`/toolsapi/toolservice/toolslog?logfile=`*`name of report from icheckerreport request`*

Action	API request
Get a list of all tools logs.	`GET https://host:port/toolsapi/toolservice/toolslog`
Run the integrity checker utility.	`POST https://host:port/toolsapi/toolservice/icheckerrepair`
Upload logs from Maximo Manage pods to S3 Cloud Object Storage.	`POST https://host:port/maximo/api/service/logging?action=wsmethod:submitUploadLogRequest`
Stop the Maximo Manage pods.	`POST http://host:port/toolsapi/toolservice/managestop`
Start the Maximo Manage pods.	`POST http://host:port/toolsapi/toolservice/managestart`
Import an external certificate to a server bundle truststore.	`POST http://host:port/toolsapi/toolservice/installexternalcert`
Validate the database for the migration to Maximo Manage.	`POST http://host:port/toolsapi/toolservice/validatedbformas`

Configuring Oracle connector after upgrade

You must define a message provider, messaging queues, and assign queues if you are using Maximo Connector for Oracle Applications for integration in Maximo Manage.

Procedure

1. Define the message provider and queues, which are Kafka or JMS.
2. Assign queues to External System OA12.
3. Configure Kafka crontask according to the MIF guide and to update settings for JMS cron task, see Configuring JMS servers.
4. If you are using Kafka, disable **JMSQSEQCONSUMER** crontask.
5. Re-create interface tables and PL/SQL objects in Oracle E-Business Suite12.x MAXORA schema.
6. Reapply customizations on PL/SQL side.
7. Enable the external system.

Configuring SAP Connector after upgrade

You must define a message provider, messaging queues, and assign queues if you are using Maximo Connector for SAP Applications for integration in Maximo Manage after you upgrade.

Procedure

1. Define the message provider and queues as either Kafka or JMS.
2. Assign queues to the external system SAP2005.
3. Configure Kafka cron task. For more information, see Integration by using Apache Kafka.
4. Update settings for the JMS cron task. For more information, see Configuring JMS servers.
5. If you are using Kafka, disable **MSQSEQCONSUMER** cron task.
6. Generate **APIKey** for integration user **sapadmin**. For more information, see Generating API keys.
7. Change connection parameters in SAP PO or Maximo HTTP channel.
8. Enable the external system SAP2005.

Checking Maximo Manage deployment status

Verify that Maximo Manage is deployed successfully by checking the **Custom Resource Definitions** page on the Red Hat OpenShift console.

Before you begin

Maximo Manage must be activated first.

Procedure

1. In the Red Hat OpenShift console, select **Administration** > **Custom Resource Definition**.
2. In the **Name** field, search for `manageworkspace` and then click the Manage Workspace custom resource definition to open it.
3. To select your project instance, select the **Instances** tab.
4. Check the project status on the **Details** tab of the **Custom Resource Definitions** page.

   The project status can be either true or false based on various conditions.

   For example, the project is build ready when the build is completed, or it is deployment ready if all the server bundles are running.

Accessing Maximo Manage

You can access Maximo Manage in different ways after you deploy it in Maximo Application Suite.

Procedure

- In Maximo Application Suite, click the **Administration** icon to open the **Suite navigator** page and on the **Applications** tab, click the **Manage** tile.
- Click the **AppSwitcher** icon and then click **Manage**.
- In Red Hat OpenShift web console, use the `ui` bundle or `all` bundle location link. Append `/maximo` to the hyperlink text and use the complete link to access Maximo Manage in a supported browser.
- Starting in Maximo Application Suite 9.1, access Maximo Manage on the side navigation menu.
  - Select Manage from **Suite** > **Administration** > **Suite** > **Applications** tab.
  - Select Manage from **Suite** > **Administration** > **Catalog** page.

What to do next

To go to Maximo Application Suite from Maximo Manage, click **IBM Maximo Application Suite** from the main menu.

Accessing database after upgrade

You might want to access the migrated database after you finish the upgrade, for example, to connect the database to a database utility tool, such as DBeaver.

Procedure

1. Log on to Red Hat OpenShift web console and from the side navigation menu, click **Networking** > **Routes**.
2. Click **Create Route**to access the Db2 host outside the cluster.
3. On the **Create Route** page, select the **YAML view** radio button and provide the YAML that includes the external hostname from the route and the external port from Services.

Tip: Search for a service name, for example, `c-db2w-manage-db2u-engn-svc` and a sample URL might be `https://ukiot-mas2-0026a8a1020f89b5eb8fa6780c129be5-0000.eu-gb.containers.appdomain.cloud`.

You can modify the following example YAML to use the values for your route:

```
kind: Route
apiVersion: route.openshift.io/v1
metadata:
  name: db2u-tls-route
  namespace: db2u
  uid: a9b76ae3-76f5-49e8-868f-ce060174936d
  resourceVersion: '114046'
  creationTimestamp: '2022-09-13T20:24:44Z'
  labels:
    formation_id: db2u-shared
  annotations:
    kubectl.kubernetes.io/last-applied-configuration: >-
      {"apiVersion":"route.openshift.io/v1","kind":"Route","metadata":
{"labels":{"formation_id":"db2u-shared"},"name":"db2u-tls-route","namespace":"db2u"},"spec":
{"host":"db2u-shared-db2u.monitordemo1-822c5cdfc486f5db3c3145c89ca6409d-0000.us-
south.containers.appdomain.cloud","port":{"targetPort":"ssl-server"},"tls":
{"nsecureEdgeTerminationPolicy":"None","termination":"passthrough"},"to":
{"kind":"Service","name":"c-db2u-shared-db2u-engn-
svc","weight":100},"wildcardPolicy":"None"}}
  managedFields:
    - manager: OpenAPI-Generator
      operation: Update
      apiVersion: route.openshift.io/v1
      time: '2022-09-13T20:24:44Z'
      fieldsType: FieldsV1
      fieldsV1:
        'f:metadata':
          'f:annotations':
            .: {}
            'f:kubectl.kubernetes.io/last-applied-configuration': {}
          'f:labels':
            .: {}
            'f:formation_id': {}
        'f:spec':
          'f:host': {}
          'f:port':
            .: {}
            'f:targetPort': {}
          'f:tls':
            .: {}
            'f:termination': {}
          'f:to':
            'f:kind': {}
            'f:name': {}
            'f:weight': {}
          'f:wildcardPolicy': {}
    - manager: openshift-router
      operation: Update
      apiVersion: route.openshift.io/v1
      time: '2022-09-13T20:24:44Z'
      fieldsType: FieldsV1
      fieldsV1:
        'f:status':
          'f:ingress': {}
      subresource: status
spec:
  host: >-
    db2u-shared-db2u.monitordemo1-822c5cdfc486f5db3c3145c89ca6409d-0000.us-
south.containers.appdomain.cloud
  to:
    kind: Service
    name: c-db2u-shared-db2u-engn-svc
    weight: 100
  port:
    targetPort: ssl-server
  tls:
    termination: passthrough
  wildcardPolicy: None
status:
  ingress:
    - host: >-
        db2u-shared-db2u.monitordemo1-822c5cdfc486f5db3c3145c89ca6409d-0000.us-
south.containers.appdomain.cloud
      routerName: default
```

```
      conditions:
        - type: Admitted
          status: 'True'
          lastTransitionTime: '2022-09-13T20:24:44Z'
      wildcardPolicy: None
      routerCanonicalHostname: >-
        router-default.monitordemo1-822c5cdfc486f5db3c3145c89ca6409d-0000.us-
south.containers.appdomain.cloud
```

4. In Maximo Application Suite, from the side navigation menu, click **Configurations**.
5. In the **Storage** section, open the **Database connection** row and view and note the database connection string, username, and password.

Troubleshooting global property values

When migrating a Maximo SaaS Flex 7.6.1.2 database, you might see an error in the **mxe.int.globaldir** property value.

Symptoms

If you have a database in Maximo SaaS Flex 7.6.1.2 and want to refresh this database to Maximo Application Suite, you might see the following error in the **mxe.int.globaldir** property value.

```
[Default Executor-thread-7] ERROR maximo.graphite - [DatabaseResourceLoader] Expand App manage-
shell failed.
Message: /opt/IBM/WebSphere/AppServer/profiles/ctgAppSrv01/MAXIMO_b1dk/maximo/maf/manage-shell/
8.0.0.0-0/app-source.zip
java.nio.file.NoSuchFileException: /opt/IBM/WebSphere/AppServer/profiles/ctgAppSrv01/
MAXIMO_b1dk/maximo/maf/manage-shell/8.0.0.0-0/app-source.zip
\tat sun.nio.fs.UnixException.translateToIOException(UnixException.java:92) ~[?:?]
\tat sun.nio.fs.UnixException.rethrowAsIOException(UnixException.java:111) ~[?:?]
\tat sun.nio.fs.UnixException.rethrowAsIOException(UnixException.java:116) ~[?:?]
```

Causes

In your database, the **mxe.int.globaldir** property value is set to something that does not exist in the image.

Resolving the problem

Make sure that the **globaldir** value is either a mounted persistent volume, or it is a directory that already exists in the image because it is used as a default system directory. If you want to temporarily use a non persistent volume by default, set the value for **mxe.int.globaldir** as blank or null.

You can also set the environment variable through the **MAF_APP_ROOT** encryption property. For example, set **MAF_APP_ROOT** to `/tmp`.

Attention: Any Maximo properties that refer to a directory or **doclink** root must be validated. If you need persistence, a Persistent Volume must be mounted to host that directory, or, point to any existing writable directory, for example, any sub-folder under `/tmp`.

Accessing Maximo Manage logs by using Red Hat OpenShift web console

View the Maximo Manage logs for Server or System.Out, MAXINST, Update database, User sync, Workspace operator, and Build from the Red Hat OpenShift console.

Procedure

1. Log into the Red Hat OpenShift console.
2. From the side navigation menu, click **Workload** > **Pods** menu.
3. Select your Maximo Manage project name from the Project drop down.
4. Select the pod from the list for which you want to view logs.

5. Click the **Logs** tab to view the streaming log for that pod.
6. To push the server logs to any S3 compatible object storage:
   a) Set up S3 credentials, by creating the following four environment variables in your Maximo Manage deployment, *LOG_BUCKETNAME*, *LOG_S3ACCESSKEY*, and *LOG_S3ENDPOINTURL*, and *LOG_S3SECRETKEY*.
   b) Create a log request, that is a POST request by using any HTTP REST Client, for example, download and use the Postman tool to create the log request.

      Request URL: `http://manageserver:7001/maximo/oslc/service/logging?action=wsmethd:submitUploadRequest`

      Note: Replace the Maximo Manage server in the URL with your Maximo Manage server host or IP address. A record for Maximo Manage is created in the LOGREQUESTDET table. The Maximo Manage records are removed by the LOGREQUESTCLEANUP cron task after the logs are posted to the object storage. The cron task must be activated manually, and then it runs once daily,
   c) Use the S3 browser to view the logs by using your S3 credentials.

      Each Maximo Manage server handles the latest log request. It compresses the log files in the log directory and uploads it to cloud object storage.

What to do next

You can install Red Hat OpenShift logging by deploying Red Hat OpenShift Elasticsearch and Red Hat OpenShift logging operators. For more information, see Installing Logging.

Verifying system settings after upgrade

After the upgrade process is complete, verify and update specific settings in Maximo Application Suite, Maximo Manage, and the Red Hat OpenShift web console.

- In the System Properties application in Maximo Manage, check that the **`mxe.int.dfltuser`** property is set to a user with administrative privileges. By default, this admin user is used to synchronize the Maximo Application Suite users to Maximo Manage.
- Check that each user has only one primary email address. Maximo Manage business logic does not allow multiple primary email addresses.
- If multiple users have the same login IDs that differ only in the usage of uppercase or lowercase letters, correct the login IDs so that they are each unique. Two users cannot have the same login IDs.
- Check that ui, report, mea, and cron routes are present in the Red Hat OpenShift cluster.
- Check that ui, report, mea, and cron services are running in the Red Hat OpenShift cluster.
- Customize the Liberty `server.xml` file, if necessary in Maximo Application Suite.
- Set the **`mxe.hostname`** property to the maximo-all or maximo-ui route host.

Updating statistics

As a system administrator, you can analyze tables in IBM Maximo Manage to ensure that the Oracle Database cost-based optimizer has up-to-date statistics.

Procedure

1. In Maximo Manage, , click **System Configuration** > **Platform Configuration** > **Database Configuration**.
2. In the **More Actions** menu, click **Update Statistics**.
3. Click **OK**.