Oracle® Cloud

Administering Oracle Mobile Cloud Enterprise in a Customer-Managed Environment

Release 18.2.3
E95891-03
August 2018

Oracle Cloud Administering Oracle Mobile Cloud Enterprise in a Customer-Managed Environment, Release 18.2.3

E95891-03

Primary Author: Catherine Pickersgill

Contents

Preface

Audience vi
Documentation Accessibility vi
Conventions vi

1 A Service Administrator’s Roadmap to Oracle Mobile Cloud Enterprise

Administering an OMCe Instance 1-3

2 Where Do I Sign Up?

3 Plan Your OMCe Instance

Logging In to Oracle Cloud Services 3-2
Do These Tasks First 3-2
Selecting a Georeplication Policy for Your Storage Service Instance 3-3
Creating a Storage Service Admin User 3-5
Update Settings for the Storage Container 3-7
Provisioning the Database Cloud Service 3-8

4 Set Up the OMCe Service

Create an OMCe Stack 4-2
Create a Bots-Only Stack 4-6
Examine Your OMCe Environment 4-9
Log In to the OMCe UI 4-12
Use IDCS to Manage Users and Roles 4-13
Add Users 4-14
Batch Import Users 4-15
Create Groups 4-17
Assign Roles 4-18

OMCe Team Member Roles 4-18
View and Edit Users 4-19
Reset User Passwords 4-20

5 Managing an OMCe Stack

View Your OMCe Stack 5-2
View Stack Activity 5-3
Start and Stop an OMCe Stack 5-3
How Do I Start or Stop a Stack with the Console? 5-3
Deleting an OMCe Stack 5-3
How Do I Delete an OMCe Stack with the Console? 5-4
Add a New SSH Public Key 5-4
Add an Access Rule 5-4

6 Patching OMCe

Typical Workflow for Patching an OMCe Stack 6-2
Configuring the Patch Script 6-2
Viewing Patch Details 6-3
Performing a Patch Precheck 6-3
Upgrading from 17.4.5 to 18.1.1 6-4
Patching OMCe Services 6-5
Patch All OMCe Services 6-5
Patch Individual OMCe Services 6-6
Upgrading Bots Samples After Patching 6-7
Troubleshoot Patch Issues 6-7
Rolling Back a Patch 6-8

7 Manage OMCe Services

Archive and Purge Bots Data 7-2

8 Troubleshoot Your OMCe Service

Troubleshoot Mobile Suite Provisioning 8-1
Analytics Not Working 8-4
Troubleshoot Service Creation Failure 8-5
Troubleshoot Runtime Mobile Suite Issues 8-6
Troubleshoot Runtime Mobile Core Issues 8-6
Using the OMCe Mobile Suite Doctor 8-6
Using the OMCe Mobile Core Doctor 8-12

Getting Log Messages 8-15

9 Managing OMCe with the PaaS CLI

Monitor Your OMCe Stack with the CLI 9-2
Start or Stop a Stack with the CLI 9-4
Delete a Stack with the CLI 9-4
Monitor a Service Instance with the PaaS CLI 9-4
Precheck Before Patching with the PaaS CLI 9-5
Patch an OMCe Service with the PaaS CLI 9-5
Rollback a Patch with the PaaS CLI 9-6

Preface

Welcome to *Administering Oracle Mobile Cloud Enterprise in a Customer-Managed Environment*.

Audience

This guide is intended for service administrators responsible for provisioning and maintaining Oracle Mobile Cloud Enterprise.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

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

Conventions

The following text conventions are used in this guide:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1

A Service Administrator’s Roadmap to Oracle Mobile Cloud Enterprise

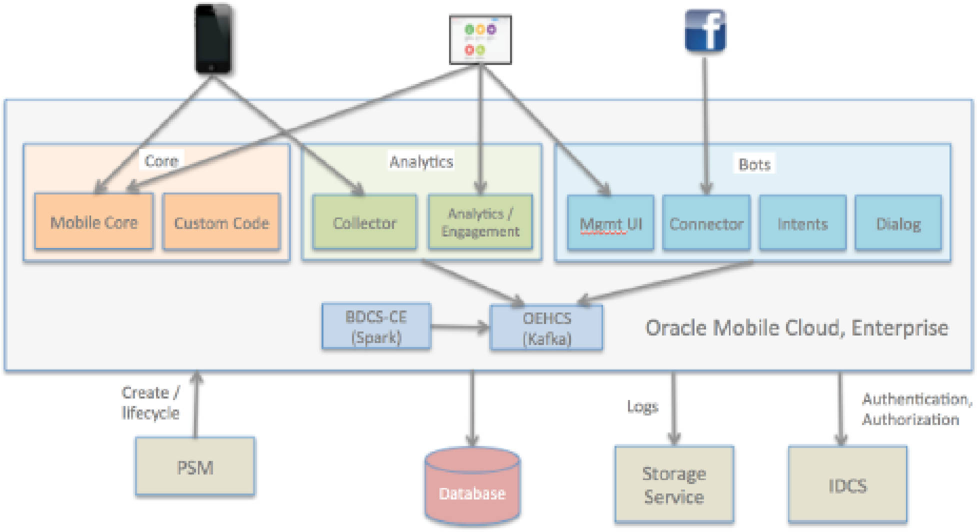
Welcome to Oracle Mobile Cloud Enterprise (OMCe)! OMCe is a cloud-based service that provides a unified hub for developing, deploying, maintaining, monitoring, and analyzing your mobile apps, intelligent bots, and the resources they rely on. As a service administrator, you can use this guide to provision, monitor, and maintain OMCe instances.

OMCe is a suite of services that consists of:

- Core - A set of integrated components that enables developers, managers, and mobile cloud administrators to develop, maintain, monitor, and export your mobile apps and the resources they rely on. The core also includes Oracle Mobile Application Accelerator (MAX), which lets you build an app from APIs and business objects.
- Analytics - Tools for analyzing, monitoring, and optimizing your mobile apps. The integrated campaign feature lets you act on insights quickly, so you can get relevant messages to the right users at the right time. With Analytics, you can create event-based user reports, funnels, segments, and retention reports to follow your users from the first click to conversion and every decision in between.
- Intelligent Bots, or chatbots - In essence, a bot is a virtual personal assistant. Bots is a conversational computing platform that enables developers to build bots (that is, apps that run automated tasks) that engage users in natural language conversations and connects them to the enterprise.

A High-Level View of an OMCe Instance

To better understand how OMCe functions, this illustration gives you a high-level view of the software topology of an OMCe instance:



The service itself is made up of the Core, Analytics, and Bots components. Requests are sent to the service from a mobile device, browser, or social messaging platform. Requests may be sent to the Core, Analytics, or Bots component, depending upon their origin.

Each component hosts its own UI but all are accessible through single sign-on (SSO), so a single user can access any or all of the component services.

- The Core component comprises the core server and the custom code server. Requests come in through the core server and if they are custom API requests they will be routed through to the custom code server, where the APIs are hosted.
- The Analytics component comprises the Ingest (or Collector) server, where all of the event requests are handled, and the Analytics-Engagement server, where some of the event processing is managed.
- The Bots component comprises the Connector, Intents, Dialog, and Management UI services. The Connector handles incoming conversation requests, while the Intents and Dialog engines do the natural language processing and manage the conversational flows. The Management UI hosts the UI application and supports the design-time browser interactions.
- Push events from Analytics, along with messages and replies from Bots, flow through a real-time streaming pipeline called Kafka. Kafka topics are provisioned in the form of a set of services called Oracle Event Hub Cloud Services (OEHCS).
- Analytics data is enriched using Spark, a part of Oracle's Big Data Cloud Service – Compute Edition (BDCS-CE). A Spark job will take the events from the Kafka topics, enrich the data, and write the enriched data out to the database.
- Data from OMCe, including metadata and analytics data, is stored in the customer’s DBCS database service. This database must be provisioned separately before provisioning the OMCe instance, and passed in as a parameter during the OMCe instance provisioning process.
- The Storage Service is used by the OMCe instance to store logs and internal metadata. Create a storage container ahead of time so you can pass it in as a parameter during the OMCe instance provisioning process.

- The PaaS Service Manager (PSM) is used to create the OMCe service instance. PSM has a Stack Manager UI that you use to access the service settings and manage the service instance lifecycle. It calls back into the individual component services to effect patching, starting, stopping and other service lifecycle events.
- Identity Cloud Service (IDCS) is used to set up user accounts and create and assign groups and roles to provide access and specific user privileges to the service. IDCS handles all authorization and authentication for the service instance.

Administering an OMCe Instance

The identity domain or service administrator is determined when subscribing to a service and setting up an account. It’s up to the service administrator or designated developer to maintain and update the provisioned instance. There might be one or more instances of OMCe to maintain. One instance could be used as a development instance while another could be used for testing, and still another instance for production. Whether one instance of OMCe is used or multiple, each instance needs to be maintained, monitored, updated, and patched.

What Do I Need to Do?

Here’s a list of the kinds of tasks you’ll need to perform to setup, start, and maintain an OMCe instance:

Task	Description	How Do I Do This?
Order an OMCe service and activate it	Purchase Universal Credits for Oracle Mobile Cloud, Enterprise. A single instance contains all the components of OMCe, including MAX, Core, Analytics, and Bots.	See Where Do I Sign Up?
Create database instance and storage instance	Provision a database instance and its associated storage container **before** activating the OMCe service.	See Provisioning the Database Cloud Service
Create the OMCe stack	Create an OMCe service instance by creating a stack.	See Create an OMCe Stack
Set up users and groups	Use Oracle Identity Cloud Service to set up Single Sign-On access to OMCe and add users and groups.	See Add Users
Monitor and manage the OMCe stack	View the status, usage, resource allocation, and other information about the OMCe service.	Managing an OMCe Stack
Patch the OMCe services	Patch an OMCe instance.	Patching OMCe
Monitor and manage the individual services within the stack	Stop an instance to apply a patch, suspend a service instance to perform maintenance tasks, and restart a service instance.	Manage OMCe Services

Task	Description	How Do I Do This?
Troubleshoot your OMCe service	Identify common errors when running a stack; learn about best practices for creating and maintaining a stack.	Troubleshoot Your OMCe Service

2 Where Do I Sign Up?

Here’s how to get started with free Oracle Mobile Cloud Enterprise promotions and subscriptions. Do one of the following:

- If you're not ready to commit to buying OMCe and just want to explore for a while, you can sign up for a trial. See Requesting and Managing Free Oracle Cloud Promotions in *Getting Started with Oracle Cloud*.
- Purchase a subscription. See Buying an Oracle Cloud Subscription in *Getting Started with Oracle Cloud*.

You will receive two emails which should arrive within a few minutes of you signing up.

- The first email has the subject *Welcome to Oracle Cloud – Infrastructure and Platform Free Promotion*. It contains your user ID and password, as well as a link to log into your Oracle Cloud account. At this point, you can log into your Cloud account and start to explore.
- The second has the subject *Your Oracle Cloud Services Are Ready - Get Started Now* and contains similar information to the first, but it also has a number of extra URLs at the bottom to Oracle Cloud Infrastructure, Ravello, and so on. You can ignore most of these URLs. In fact, to provision an OMCe instance you MUST use the default `MyServices` URL, which is the one that starts with `https://myservices-cacct-…`. Do not use any of the other account URLs.

3

Plan Your OMCe Instance

Before you begin setting up an Oracle Mobile Cloud Enterprise(OMCe) instance, you should think about what you are going to use OMCe for and make a number of decisions, such as whether you need a database with Enterprise or High Performance packaging.

When you start to plan your OMCe instance, these are some of the things to consider:

- Do you want the full OMCe stack, which includes Bots and Analytics, or do you want a Bots-only stack? If you only want to work with Bots, you can create the instance using the Bots-only template.
- Are you are trying out OMCe to produce pilot applications or as proof of concept, or are you setting up a production environment? The database and OCPU requirements are different for each.
- How are you going to manage the conversation history that Bots will be writing to the database over time? Do you have a plan to archive and purge the data so that database limits aren’t breached. See Archive and Purge Bots Data.

Provisioning the Database for OMCe for Proof of Concept

If you want to try out OMCe and examine the features available and understand the concepts, and perhaps develop some trial applications, you can use these recommendations.

For a full OMCe stack you need:

- DB requirements (type and OCPU): Enterprise Package minimum
- OCPU for Oracle Event Hub Cloud Service: 1 for each environment
- OCPU for Oracle Big Data Cloud Service: 2 for each environment

For Bots-only you need:

- DB requirements (type and OCPU): Standard Package minimum
- OCPU for Oracle Event Hub Cloud Service: 1 for each environment

Provisioning the Database for a Production OMCe Environment

When you are confident that you want to create an OMCe environment suitable for a production environment, use these recommendations.

For a full OMCe stack you need:

- DB requirements (type and OCPU): Enterprise Package minimum, although if you’re planning to work with larger sets of analytics data then you should ensure that you start with a High Performance database that can handle analytics queries across larger data sets.
- OCPU for Oracle Event Hub Cloud Service: 1 for each environment
- OCPU for Oracle Big Data Cloud Service: 2 for each environment

For Bots-only you need:

- DB requirements (type and OCPU): Enterprise Package minimum
- OCPU for Oracle Event Hub Cloud Service: 1 for each environment

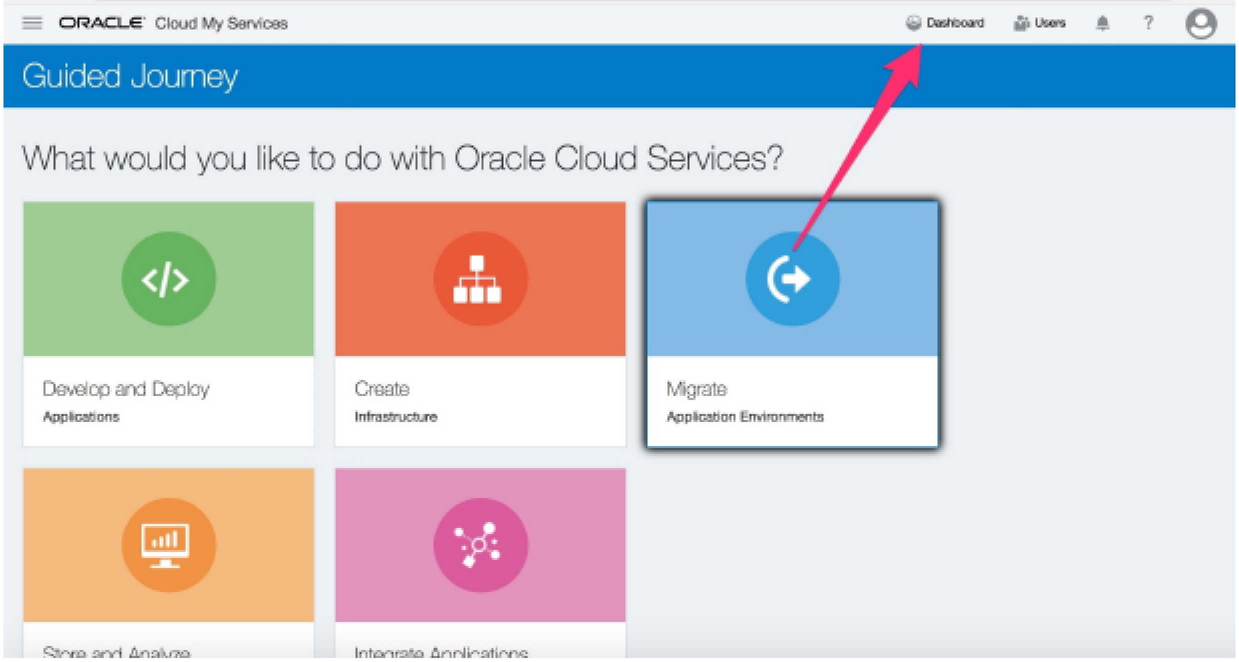
Logging In to Oracle Cloud Services

You use the Oracle Cloud MyServices applications dashboard to create and manage the OMCe stack and the services associated with it.

To log in:

1. In the email with the subject, *Your Oracle Cloud Services Are Ready - Get Started Now*, click the My Services Administration link. This is the one that starts with `https://myservices-cacct-….`
2. Sign in using the credentials in the email with the subject *Welcome to Oracle Cloud – Infrastructure and Platform Free Promotion*. If this is the first time you have signed in, you must set up a new password and the new password must match the password criteria that you'll see listed.

The first time that you log in, you are taken to the Guided Journey page, which lets you explore different Oracle Cloud Services. Because you know you want to provision Oracle Mobile Cloud Enterprise, you can ignore the Guided Journey for now and click **Dashboard** in the header to access the MyServices dashboard.



Do These Tasks First

Before you begin setting up an Oracle Mobile Cloud Enterprise (OMCe) instance, you have to provision the database and storage instances and carry out a few other tasks.

Important:

There are some tasks you must perform before you can create an OMCe stack:

1. If this is the first time you've created a service on Oracle Public Cloud, choose a georeplication policy.
2. Create a Storage Service admin user.
3. Provision Database Cloud Service, if needed.

Important Pre-Provisioning Tasks

Use this workflow as a guide to perform the tasks you must carry out before you can create and provision OMCe.

Task	Description
Set a georeplication policy for the storage service instance	If this is a new Cloud account and you've never set up a service, you must select a georeplication policy. If you've already set up at least one service, you can skip ahead to provisioning the database. Choose a georeplication policy that defines your primary data center and also specifies that your data should be replicated to a geographically distant (secondary) data center. See Selecting a Georeplication Policy for Your Storage Service Instance
Create a storage service admin user and update storage container settings	Create a Storage Service admin user for all the services you create. See Creating a Storage Service Admin User and Update Settings for the Storage Container
Create a database service and a storage service	From the Oracle Cloud My Services application, create a database service and specify the storage service to be used for backup. See Provisioning the Database Cloud Service

After you have carried out these tasks, the next step is to create the OMCe stack.

Selecting a Georeplication Policy for Your Storage Service Instance

If this is a brand new Cloud account and you've never set up a service, you must select a georeplication policy.

About Georeplication

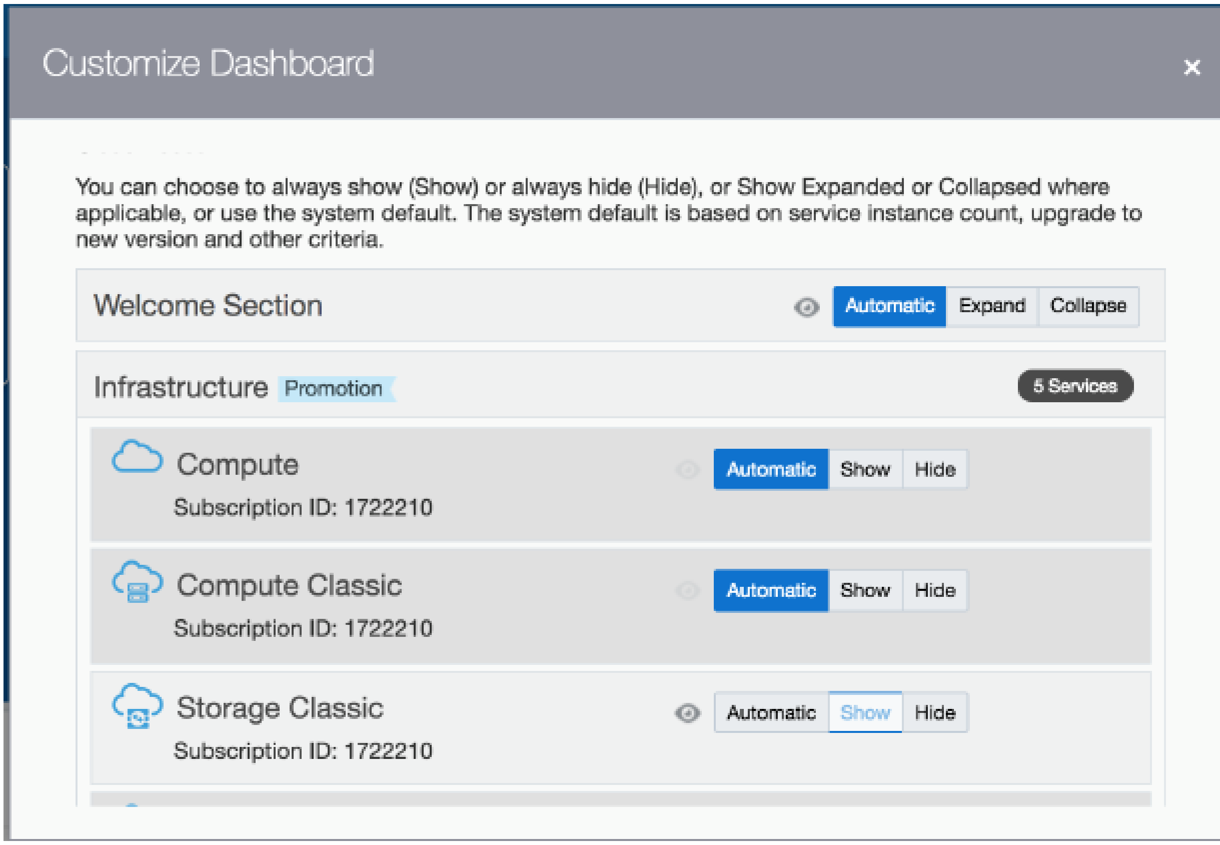
You must choose a georeplication policy that defines your primary data center—which you can find in your Welcome email—and also specifies that your data should be replicated to a geographically distant (secondary) data center. Data is written to the primary data center and replicated asynchronously to the secondary data center. In addition to being billed for storage capacity used at each data center, you'll also be billed for bandwidth used during replication between data centers.

> **Note:**
>
> Once you select a replication policy, you can't change it, so select your policy carefully.

Selecting a Georeplication Policy

1. From the MySerivces page, first select **Customize Dashboard**, then select **Show** for the Storage Classic service:
2. On the Dashboard, look for **Storage Classic**, and from ☰ select **Open Service Console**.

   This takes you to the Storage Services console. If you haven't set a georeplication policy yet, you will be prompted to do so.



Customize Dashboard

You can choose to always show (Show) or always hide (Hide), or Show Expanded or Collapsed where applicable, or use the system default. The system default is based on service instance count, upgrade to new version and other criteria.

Welcome Section — Automatic | Expand | Collapse

Infrastructure Promotion — 5 Services

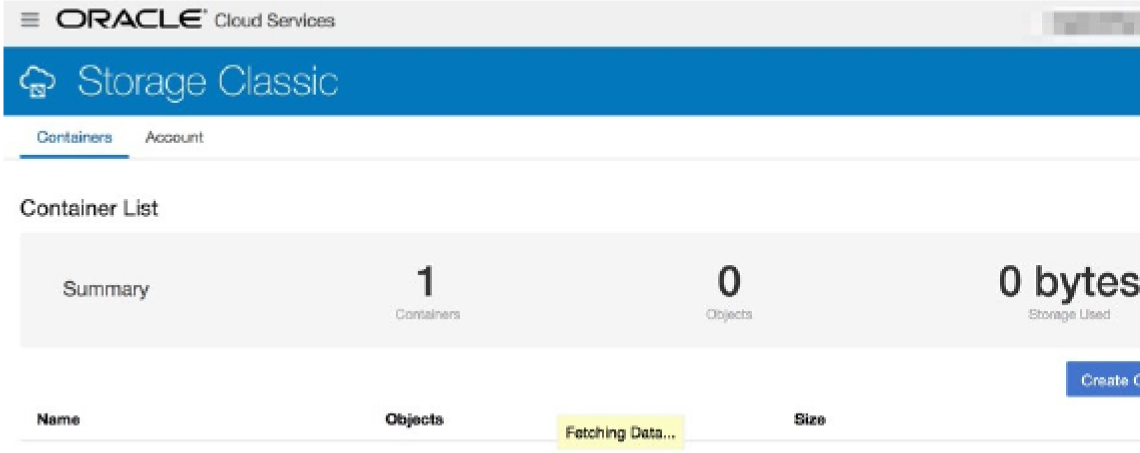
Compute — Subscription ID: 1722210 — Automatic | Show | Hide

Compute Classic — Subscription ID: 1722210 — Automatic | Show | Hide

Storage Classic — Subscription ID: 1722210 — Automatic | Show | Hide

   For trials, the default value is sufficient. For paying accounts, see About Replication Policy for Your Account in *Using Oracle Cloud Infrastructure Object Storage Classic* for an explanation of the policies. You may not see all the policies described in the documentation, as availability is based on account type and service location. Selecting a policy can be confusing, but the main decision point is whether you need data replication (replication of files between data centers in the same region) or not. Only the policies that are available for your account type and service location are shown.

   Choose the replication policy based on your needs.
3. When you see the Storage Classic overview page, you've finished this procedure.



At a later date, you can see the georeplication policy that’s selected for your storage service instance. Sign in to the MyServices Dashboard. Expand **Account Information**. The details of your account are displayed in the **Account Information** pane. Look for the **Georeplication Policy** field.

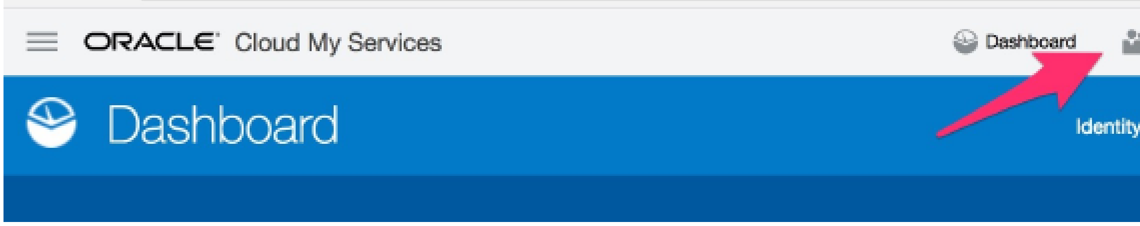
Creating a Storage Service Admin User

Because storage service admin credentials are requested every time a service needs to access storage, we recommend that you create a dedicated storage admin to use across all services. The admin user’s credential is accessed very frequently—in fact, any time any service needs to access storage, the credential is required to authenticate. The credential (token) is also cached, so if you change this user’s password, storage access will actually fail, and the account will be locked.

> **WARNING:**
>
> **After you use this storage admin user to create a service, do not change the password! Doing so will lead to runtime failures for these services.**

To create a dedicated storage service admin user:

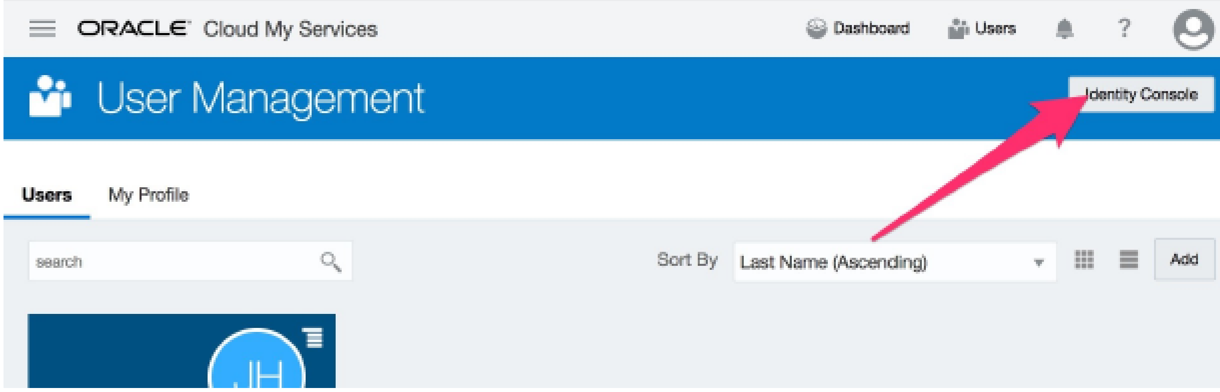
1. Click **Users** on the upper right corner of the MyServices Dashboard:



2. On the Dashboard, look for **Storage Classic**, and from ☰ select **Open Service Console**.

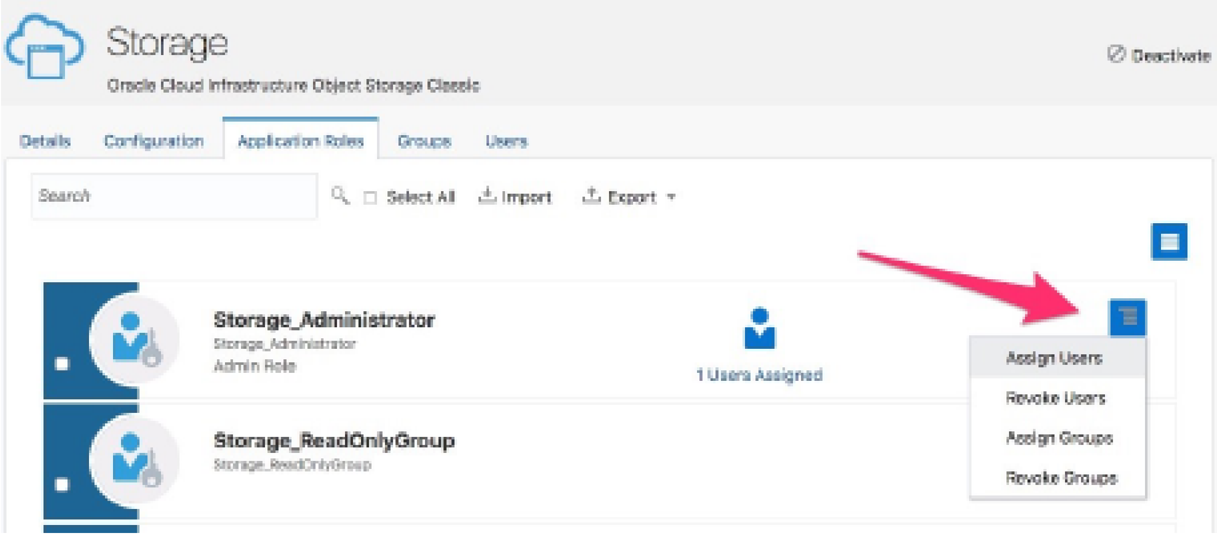
   This takes you to the Storage Services console. If you haven’t set a georeplication policy yet, you will be prompted to do so.

3. In the User Management page, click Identity Console.



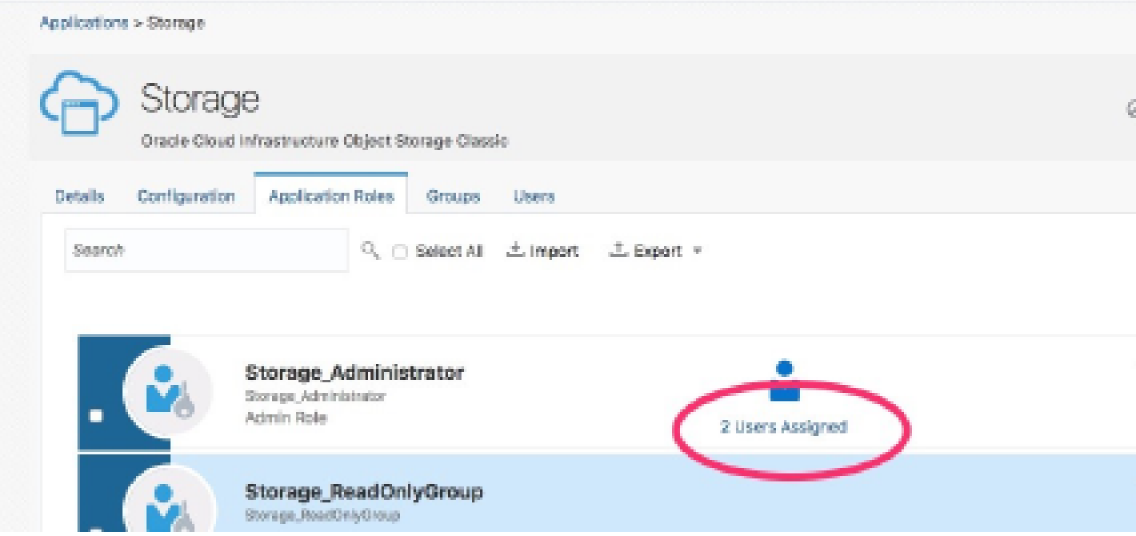
4. On the Identity Console's **User** tab, click **Add**.
5. Fill out the User Details. You need to use a different email address from your admin ID. Click **Finish** when complete.
6. Click the **Applications** tab.
7. In the search box, type **Storage**, then click **Enter**.

   You should now see the Storage service entry.
8. Click the storage service entry, then click the **Application Roles** tab.
9. Click the hamburger icon in the **Storage_Administrator** role entry, then click **Assign Users:**



10. Select the checkbox next to the newly created user, and click **Assign**.

    The new user is assigned the Storage Service administrator role, and they will receive an email with instructions to activate the account. Make sure the account is activated before you proceed. You can test this by simply logging in using this account.



Update Settings for the Storage Container

After you create the OMCe stack you must run commands to set the replication policy for storage account and set the storage container to allow anonymous read.

Use the cURL command-line tool to run commands to update settings for the storage container.

Install cURL

These steps describe the procedure on a Windows 64-bit system.

1. In your browser, navigate to the cURL home page at http://curl.haxx.se and click **Download** in the left navigation menu.
2. On the cURL Releases and Downloads page, locate the SSL-enabled version of the cURL software that corresponds to your operating system, click the link to download the ZIP file, and install the software.
3. Navigate to the cURL CA Certs page at http://curl.haxx.se/docs/caextract.html and download the **ca-bundle.crt** SSL CA certificate bundle in the folder where you installed cURL.
4. Open a command window, navigate to the directory where you installed cURL, and set the cURL environment variable, `CURL_CA_BUNDLE`, to the location of an SSL certificate authority (CA) certificate bundle. For example:

   ```
   C:\curl> set CURL_CA_BUNDLE=ca-bundle.crt
   ```

There are four commands that need to be run. Run them in the order shown.

Storage Container - Setup Replication Policy

Run these commands to set the replication policy for storage account.

get token

```
curl -v -X GET -H 'X-Storage-User:Storage-<Cloud Account>:<User>' -H 'X-Storage-
Pass:<Password>' <Storage URL>/auth/v1.0
```

Example

```
curl -v -X GET -H 'X-Storage-User: Storage-mjondcm208:mary.jones@greencorp.com' -H
'X-Storage-Pass: welcome1' https://foo.storage.oraclecloud.com/auth/v1.0
```

set policy

```
curl -v -X POST -H 'X-Auth-Token: <AUTH_Token>' -H 'X-Account-Meta-Policy-
Georeplication: <Data Center>' <Storage URL> /v1/Storage-<Cloud Account>
```

Example

```
curl -v -X POST  -H 'X-Auth-Token:AUTH_tka2f992ca45781566f48f7e7332a92c77' -H 'X-
Account-Meta-Policy-Georeplication: dc1' https://foo.storage.oraclecloud.com/
MyService-bar
```

Storage Container - Set Anonymous Read

The storage container needs to set the anonymous read flag

set read flag

```
curl -v -X POST -H 'X-Auth-Token: <AUTH_Token>' -H 'X-Container-
Read: .r:*,.rlistings' <Storage URL> /v1/Storage-<Cloud Account>
```

Example

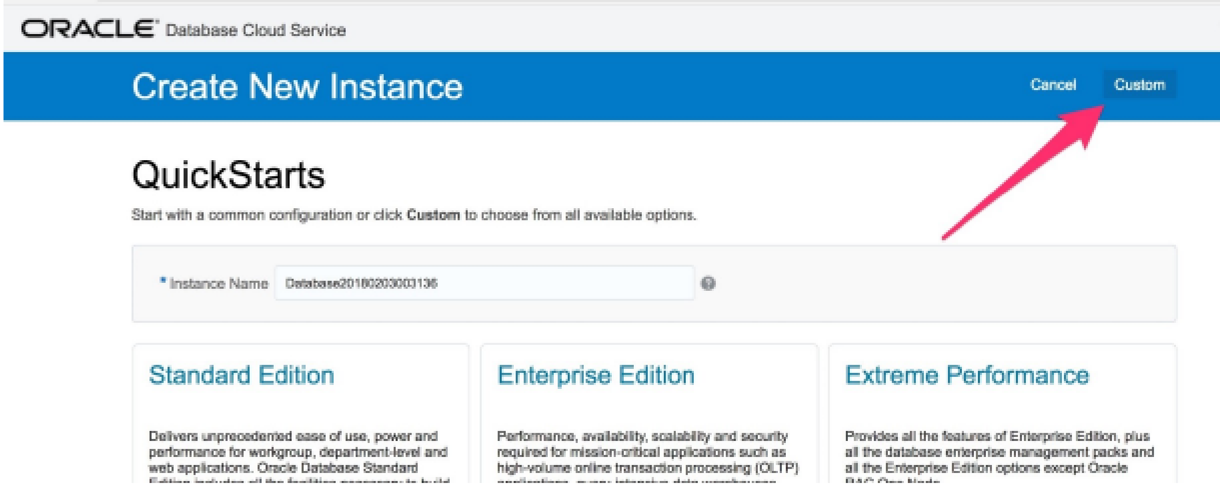
```
curl -v -X POST  -H 'X-Auth-Token:AUTH_tka2f992ca45781566f48f7e7332a92c77' -H 'X-
Container-Read: .r:*,.rlistings' https://foo.storage.oraclecloud.com/MyService-bar
```

Provisioning the Database Cloud Service

If you have already provisioned a Database Cloud Service and wish to use the same instance for OMCe, skip ahead to Set Up the OMCe Service. You'll need to supply some database-related details when you get to the OMCe provisioning step, so make sure you have that information handy.

Database Cloud Service creation typically takes 1 to 2 hours.

1. Go to the MyServices dashboard.
2. Click **Create Instance**.
3. Click the **Create** button next to **Database**.
4. On the QuickStarts page, click **Custom** in the upper right corner to create a Database Cloud Service with custom configuration:



Start with a common configuration or click **Custom** to choose from all available options.

* Instance Name: Database20180203003136

Delivers unprecedented ease of use, power and performance for workgroup, department-level and web applications. Oracle Database Standard

Performance, availability, scalability and security required for mission-critical applications such as high-volume online transaction processing (OLTP)

Provides all the features of Enterprise Edition, plus all the database enterprise management packs and all the Enterprise Edition options except Oracle

5. In the **Create Instance** dialog, fill in the provisioning configuration details:

   a. **Service Name**: Name of the DBCS instance you're creating. Choose a name that reflects its usage. The name has to be unique. It can be up to 50 characters, although it's best to use a short name. It must start with a letter, and can contain only letters, numbers and hyphens (-). It cannot end with a hyphen (-).

   b. **Description**: Enter the purpose of this DBCS instance.

   c. **Notification Email**: This is the email account receiving DBCS instances event notifications. Enter the Admin's email address.

   d. **Region**: This is the data center/region of the DBCS instance. See About Replication Policy for Your Account in *Using Oracle Cloud Infrastructure Object Storage Classic* for a mapping between data center code (e.g. `uscom-central-1`) and the actual location. The **Region** list contains only a subset of all data center codes. Select the data center location based on what you set up in the georeplication policy.

      > **Note:**
      >
      > When you provision OMCe, you must select the same region that you select here.

   e. **Bring Your Own License**: If you are setting up a trial environment, select this to help save free trial credits.

      You may activate the Bring Your Own License (BYOL) version of OMCe and you will be charged the BYOL rate for the activated OMCe provided that you have sufficient supported on-premises licenses as required and specified in the Service Description for Oracle PaaS. See Overview of Oracle Cloud Subscriptions in *Getting Started with Oracle Cloud*.

   f. **Software Release**: The version of DBCS. Select **Oracle Database 12c Release 2**. Do not accept the default value.

   g. **Software Edition**: The edition of DBCS. Choose either **Enterprise Edition - Enterprise** or **Enterprise Edition – High Performance**. For most situations, an Enterprise Edition database is sufficient. High Performance Edition is recommended only if:

      - You plan to use Mobile/Bot Customer Analytics features, and
      - There will be a large amount of historical data.

      See Plan Your OMCe Instance.

   h. **Database Type**: Leave as the default **Single Instance**.

6. Click **Next** and fill in database configuration information:

   a. **DB Name**: Enter a name for the DB instance.

   b. **PDB Name**: The name of the PDB. Use the default value.

   c. **Administration Password** and **Confirm Password**: The password for **SYS** and **SYSTEM** database users, the admin Oracle GlassFish Server user, and the admin Oracle Application Express user.

> **Note:**
>
> The password must be alpha numeric and can contain the special characters $, # , _. However, the password must not start with a special character or a number, otherwise provisioning of some services will fail.

   d. **Usable Database Storage**: For OMCe, the storage service uses DB Storage, as they are stored as BLOBs. Leave as the default of `25GB`.

   e. **Compute Shape**: The amount of resources allocated to this DBCS instance. See Classic Compute Shapes Available.

      Use the default, `OC3`.

   f. **SSH Public Key**: The public and private key pairs to use for DBCS SSH access. If you created a key pair previously, you can use it.

      To create a new public and private key pair:

      i. Click **Edit**, then **Create a New Key**. A public and private key pair are created.

      ii. Click **Download** to download the key. Save the key for later use.

      iii. Enter the public key in **SSH Public Key**.

7. Provide the information for backup and recovery:

   a. **Backup Destination**: The destination of the DBCS backup. Select `Cloud Storage Only`.

   b. **Cloud Storage Container**: Use the default value for the Cloud Storage Container name for your service instance backups. It has the format `https://<account_name>.<country>.storage.oraclecloud.com/v1/Storage-<account_name>/DBaaS-DBname`, where `DBaaS-DBname` is the container name. If you are recreating an OMCe stack, use a new container name.

      The container name must consist of UTF-8 characters only. The name can start with any character and can't exceed 1061 bytes. Don't include a slash (/), angled brackets (<>) , or strings (/../) in the name.

   c. **User Name**: This is the Storage Cloud Service Admin ID. Use the storage service user ID created in Creating a Storage Service Admin User. Do not use a real user's ID.

   d. **Password**: Use the password for the storage service admin user.

   e. **Create Cloud Storage Container**: Select this to create the Cloud Storage Container.

8. Specify whether to initialize data from backup.

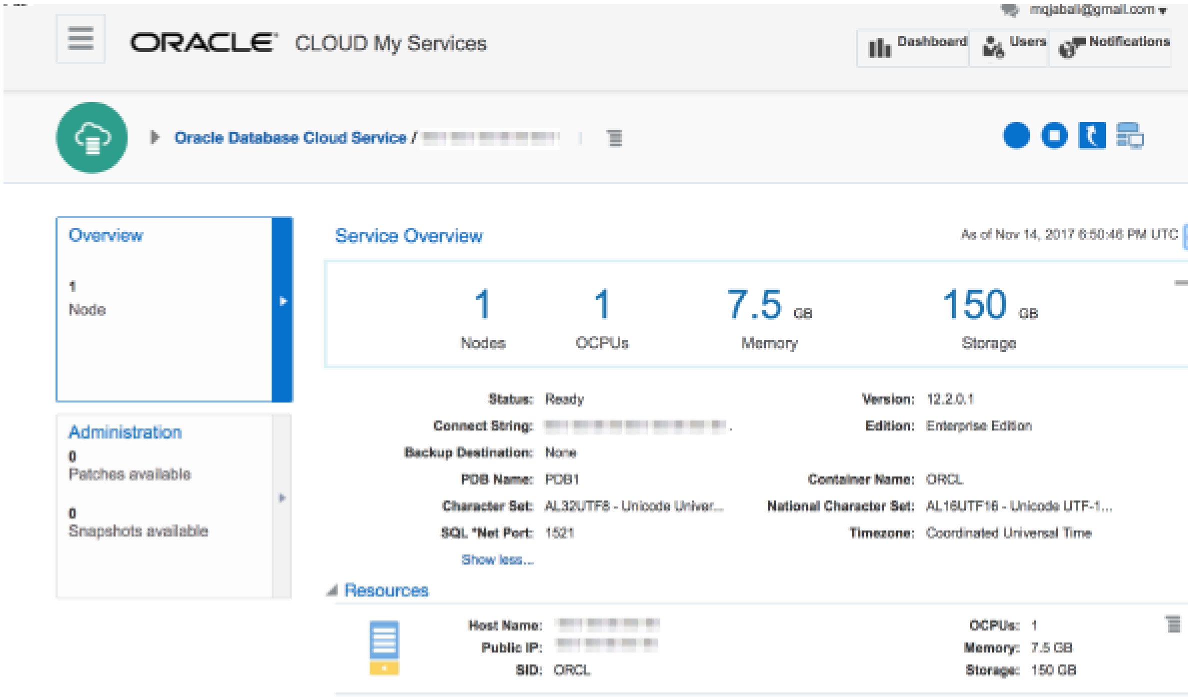
   Create Instance from Existing Backup: Specify whether the instance should be created from an existing backup. Choose `No`, unless you are really restoring from a previously DBCS instance.

9. Click **Next** and review the information you provided.

10. Click **Create** to kick off the database creation process.

On the Database Cloud Service page, you'll see your new database instance listed. When you see **Creating instance …**, you know that DBCS provisioning is taking place. DBCS creation typically takes 1-2 hours.

Once provisioned, you will see the service overview for DBCS. It should be started and ready to go.



Overview
1
Node

Administration
0
Patches available
0
Snapshots available

Service Overview

As of Nov 14, 2017 6:50:46 PM UTC

1	1	7.5 GB	150 GB
Nodes	OCPUs	Memory	Storage

Status: Ready
Connect String: [redacted]
Backup Destination: None
PDB Name: PDB1
Character Set: AL32UTF8 - Unicode Univer...
SQL *Net Port: 1521
Version: 12.2.0.1
Edition: Enterprise Edition
Container Name: ORCL
National Character Set: AL16UTF16 - Unicode UTF-1...
Timezone: Coordinated Universal Time

Resources

Host Name: [redacted]
Public IP: [redacted]
SID: ORCL
OCPUs: 1
Memory: 7.5 GB
Storage: 150 GB

4 Set Up the OMCe Service

Here’s what you need to know to get your Oracle Mobile Cloud Enterprise (OMCe) service instance set up. You’ll learn how to create an account, activate the service, create a service instance, and assign team members.

Go through each set up task in this section carefully to ensure that you have fully provisioned the service for what your team needs.

Make sure you have the following information from setting up the Database and Storage instances. You should be able to find most these from the Oracle Cloud My Services dashboard:

- The Cloud Storage Service URL
- The Cloud Storage username and password
- The service name for the database
- The administrator password used by **SYS** and **SYSTEM** database users

Provisioning Workflow

Use this workflow as a guide when provisioning an OMCe instance from scratch.

Task	Description
Create an OMCe stack	Create the OMCe stack from the OMCe template, which is the blueprint for coordinating the creation of the service and the resources it uses. It can take a few hours to provision a stack.
Add users to the OMCe service	Connect directly from the Stack Manager to IDCS to create user accounts. You can also use scripts to batch import user accounts.
Create users and assign them to roles	Create groups and assign users to them.

Before You Start

Before creating the OMCe stack, ensure that the DBCS is provisioned, up and running. See Provisioning the Database Cloud Service. If this has not been done, complete it before moving ahead.

Finding the Storage Service URL

In most cases the storage service associated with the Oracle Cloud account you are using to create the OMCe stack is the one you want to use. However there may be occasions when you want to create an OMCe stack which uses a storage cloud service in a different cloud account. In this case, you will need to have the URL for the storage service ready when you create the OMCe stack.

This URL is a combination of the URL for your storage service, plus the storage container name you specify to store artifacts. In other words, <*Cloud Storage Service URL*>/<*Name you assign to the OMCe storage container*>.

To get the correct Storage Service URL:

1. Sign in to the **Oracle Cloud My Services** application using the credentials for the Oracle Cloud account it belongs to.
2. Find **Storage Classic** on the Dashboard. If it's not on the dashboard, click **Customize Dashboard**, then click **Show** next to the **Storage Classic** entry, and close the Customize Dashboard window.
3. Click the **Storage Classic** block. **Do not** click on the hamburger icon ☰ to get to Storage Classic.
4. Scroll all the way down in the Service: Oracle Cloud Infrastructure Object Storage Classic view. In the **Additional Information** section, find the **REST Endpoint**field:
5. Record the value of the **REST Endpoint** field. Typically, this value has the format `http://<` *`Account Name`*`>.storage.oraclecloud.com/v1/Storage-<`*`Account Name`*`>`.
6. Decide on the Storage Container Name; for example, `dev1`.

Example: Suppose your Cloud account name is `mycloud` and is provisioned in a US data center. Your Storage Service REST endpoint would typically be: `https://mycloud.us.storage.oraclecloud.com/v1/Storage-mycloud`. To get the Storage Container URL you need to provision OMCe, you simply add the storage container name to the end of the endpoint, as in: `https://mycloud.us.storage.oraclecloud.com/v1/Storage-mycloud/dev1`.

> **Note:**
>
> Be sure you have the correct value of the **REST Endpoint** field in hand before you try to provision OMCe. Provisioning frequently fails because this value is incorrect.

Now you're ready to create the OMCe stack. Each OMCe stack is a standalone OMCe environment, so if you need dev, test, and production environments, you'll need to create three stacks.

Create an OMCe Stack

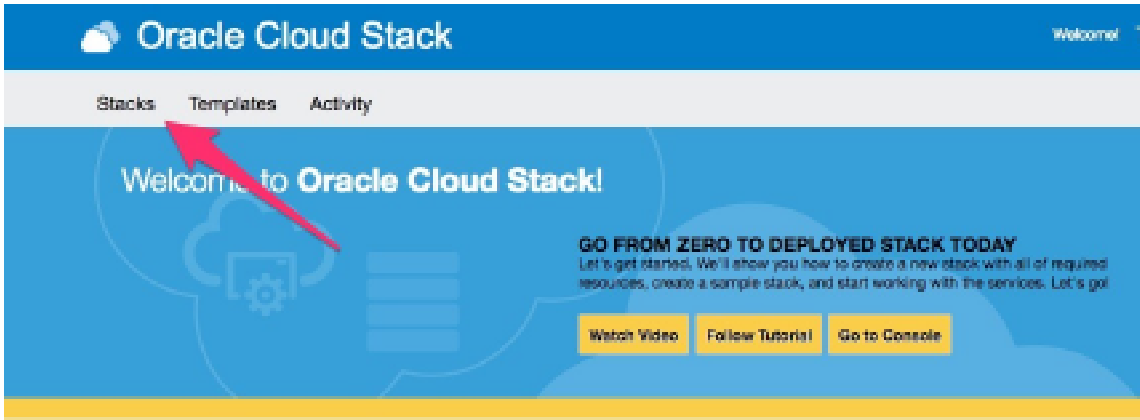
OMCe uses **service instances** (also referred to as *environments*) to define the behavior of artifacts and control access to development and administrative features. As a service administrator, you'll create an OMCe service stack , assign predefined team member roles, and configure environment policies.

A *cloud stack* is a collection of related cloud services that you've provisioned. A single OMCe stack represents a complete OMCe environment (OMCe core, custom code components, Analytics, and Bots). When you create a service instance with the Oracle Cloud Stack Manager, it provisions the Oracle Cloud resources by using each service's public REST APIs. A cloud stack is created from a template. You'll use the OMCe template, which is a blueprint for coordinating the creation of the service and the resources it uses. After you create the stack, use the Stack Manager to view and monitor the performance of the OMCe service instance.

Note:

Although you can make changes to the template for some Oracle Cloud services, it is important that you do not make any changes to the `Oracle-Mobile-Cloud-Enterprise-Template` template.

1. Sign in to MyServices dashboard.
2. Click **Create Instance**, then click the **All Services** tab.
3. Scroll down until you find the **Application Development** category, then locate **Mobile**.
4. On the My Services page, click **Stacks**:



Welcome to Oracle Cloud Stack!

GO FROM ZERO TO DEPLOYED STACK TODAY

Let's get started. We'll show you how to create a new stack with all of required resources, create a sample stack, and start working with the services. Let's go!

5. On the Oracle Cloud Stack Stacks page, click **Create Stack**.
6. You'll be prompted with the Create New Instance screen. In the **Template** field, select **Oracle-Mobile-Cloud-Enterprise-Template**. You should now see the Oracle Mobile Cloud Enterprise Stack Creation screen.
7. Fill in the fields as follows. For the Stack Details section:
   - **Name**: Enter a name that reflects usage of the environment, for example, for a development environment, call it `dev1`.

     You can use a name of up to 20 characters. It must start with a letter, and can contain only letters, numbers and hyphens (-). It cannot end with a hyphen (-).

     Use a different name for each environment you create. Also, use a different name from previous attempts, if you are re-running a failed deployment, even if the previous attempts have been completely rolled back.
   - **Description**: Enter a brief description of the stack that describes what the stack is used for.
   - **Notification email**: The notification email address comes from the log in account. Status reports are sent to this email.
   - **Tag**: (Optional) Add tags and assign tags to this service.
   - **On Failure Retain Resources**: Controls whether to rollback resource creation on failure or not. It is recommended that you **check** this, as it will allow you to resume a partially completely OMCe provisioning process, which could greatly speed up provisioning re-try.

8. Complete the Access section:
   - **SSH Public Key**: The Public Key value. You can re-use the DBCS SSH key or create a new one. If your DB and OMCe administrators are different, you'll want to use different keys.

     Only use the keys generated from your Oracle Cloud account. Avoid using keys generated using external tools.
   - **SSH Private Key**: Browse to the Private Key file generated with the public key.
   - **Administrator Username**: The default is `omceadmin`. You can change this if you wish.

     The name must have no fewer than eight characters and cannot exceed 128 ASCII characters. You cannot use a tab, brackets, parentheses or the characters , < > & # | ?.
   - **Administrator Password**: Enter a password for the administrator and re-enter it to confirm.

     This password is used across all suite components for their respective administrative users, for example, the WLS admin user. The password must have no fewer than eight characters and cannot exceed 30 characters. There must be at least one lower case letter, one upper case letter, one number, and one special character, either _ or #, for example, `Ach1z0#d`.
9. For Storage Container Configuration:
   - **Change Default Cloud Storage Service URL**: Leave unchecked unless you want to use a different storage cloud service in a different cloud account. This is something that is only rarely needed.

     If you do want to use a storage cloud service in a different cloud account, select **Change Default Cloud Storage Service URL** and enter the Cloud storage service URL. See Set Up the OMCe Service.
   - **Cloud Storage User Name**: Enter the name of the Storage Cloud Service admin user you created earlier. This user should only have the Storage Cloud Service admin role, and nothing else.
   - **Cloud Storage Password**: Enter the password for the Cloud Storage admin user.
10. For Oracle Mobile Cloud Enterprise Configuration:
    - **Hourly API Request Capacity**: The number of requests per hour you want your OMCe environment to handle.

      OMCe metering/billing will be based on this number, regardless of actual usage. In other words, if this value is set to 2500 requests per hour, then OMCe will bill 2500 * OMCe price per hour (Say 0.0021) = $5.25 per hour. This will continue regardless whether there is any actual traffic/requests through OMCe.

      > **Note:**
      >
      > Carefully review this number and do not accept default value. 2500 is considered high volume.

For Dev/Test/Production environments, this is an average value of the requests per hour. Take the expected number of requests for the entire month, divide by 744, then enter this value.

For trials, you can enter a relatively low number, like 10 or 50 requests per hour.

Note that OMCe will continue to operate normally regardless of whether your actual usage is higher or lower than the value specified.

- **Region**: Defines the data center within your data region where the OMCe instance will be provisioned. Select the data center location based on your actual location.

  > **Note:**
  >
  > This is a mandatory field. You must specify a region. If you don't, OMCe components may be provisioned to different data centers which would result in provisioning failure.
  >
  > The region you select must be the same one that you specified for the database service.
  >
  > Do not use the default `No Preference`.

  For a mapping between data center code (e.g. `uscom-central-1`) and the actual location, see About the Replication Policy for Your Account in *Using Oracle Cloud Infrastructure Object Storage Classic*.

11. For Database Access:

    - **Database Cloud Service Name**: Name of the DBCS instance used for this OMCe stack/environment. This DBCS instance can be shared across multiple OMCe environments. Enter the name of the DBCS instance you want to use with this OMCe stack.
    - **PDB — Pluggable Database Name**: Accept the default of `PDB1`.
    - **DBA Password**: Enter the password you used when you created the DBCS. This password can't start with a special character.
    - **Schema Prefix**: All schema objects (i.e. tables, etc.) are prefixed with this string to identify its use. Use a different prefix for each environment you create. If you are re-running a failed deployment, you must also use a new prefix, even if the previous attempts have been completely rolled back.

      Must be all upper case letters and numbers, with a maximum of 11 characters. It must be unique.
    - **Force Schema Creation**: Do not select this.
    - **Enable High Performance Database Features for Advanced Analytics**: If DBCS High Performance or Higher Editions are provisioned and are being used for this OMCe stack, check this box. If you only have Enterprise Edition or below, leave this box unchecked

12. Before you proceed, carefully review the requirement for the schema prefix which must be in all uppercase. Most important, be sure to use a different prefix for each OMCe environment/instance you create. If you are re-running an OMCe instance

creation that previously failed, you must use a different schema prefix to avoid unexpected naming conflicts.

13. When you have finished entering the details, click **Next**.
14. Click **Confirm**.

OMCe stack creation takes about 1-2 hours, and creates 22 resources. You can follow progress in the status pane which displays a list of the instances as they are created. You can click the name of the service to find more information.

When the stack creation is complete, you'll get a confirmation email.

Create a Bots-Only Stack

If you only want to use Bots and you are not intending to use Mobile Core or Analytics you can create a Bots stack. Although this has the benefit of reducing resource consumption, you will ONLY get the Bots infrastructure—neither Core nor CxA will be provisioned. This means that you must host custom components, which are required for bots, in a separate ACC container, rather than in mobile core. If you subsequently decide that you do want to use Mobile Core and Analytics you will have to create a new stack. You cannot change a Bots stack into a full OMCe stack.

> **Note:**
>
> Although you can make changes to the template for some Oracle Cloud services, it is important that you do not make any changes to the `Oracle-Mobile-Cloud-Enterprise-Template` template.

1. Sign in to MyServices dashboard.
2. Click **Create Instance**, then click the **All Services** tab.
3. Scroll down until you find the **Application Development** category, then locate **Mobile**.
4. On the My Services page, click **Stacks**.
5. On the Oracle Cloud Stack Stacks page, click **Create Stack**.
6. You'll be prompted with the Create New Instance screen. In the **Template** field, select **Oracle-Mobile-Bots-Standard-Rel**. You should now see the Oracle Bots Stack Creation screen.
7. Provide information for the Stack Details:
   - **Name**: Enter a name that reflects usage of the environment, for example, for a development environment, call it `dev1`.

     You can use a name of up to 20 characters. It must start with a letter, and can contain only letters, numbers and hyphens (-). It cannot end with a hyphen (-).

     Use a different name for each environment you create. Also, use a different name from previous attempts, if you are re-running a failed deployment, even if the previous attempts have been completely rolled back.
   - **Description**: Enter a brief description of the stack that describes what the stack is used for.

- **Notification email**: The notification email address comes from the log in account. Status reports are sent to this email.
- **On Failure Retain Resources**: Controls whether to rollback resource creation on failure or not. It is recommended that you **check** this, as it will allow you to resume a partially completely OMCe provisioning process, which could greatly speed up provisioning re-try.
- **Tag**: (Optional) Add tags and assign tags to this service.

8. Complete the Database Configuration section:
    - **Database Cloud Service Name**: Name of the DBCS instance used for this OMCe stack/environment. This DBCS instance can be shared across multiple OMCe environments. Enter the name of the DBCS instance you want to use with this OMCe stack.
    - **DBA Username**: Enter the name of the DBCS user for this DBCS instance.
    - **DBA Password**: Enter the password you used when you created the DBCS. This password can’t start with a special character.
    - **Schema Prefix**: All schema objects (i.e. tables, etc.) are prefixed with this string to identify its use. Use a different prefix for each environment you create. If you are re-running a failed deployment, you must also use a new prefix, even if the previous attempts have been completely rolled back.

      Must be all upper case letters and numbers.

      > **Note:**
      >
      > Do not use a very long schema prefix name otherwise stack creation may fail after it has been running for some time.

    - **Force Schema Creation**: Do not check this.

9. For Oracle BOTS Cloud Service Configuration:
    - **SSH Public Key**: The Public Key value. You can re-use the DBCS SSH key or create a new one. If your DB and OMCe administrators are different, you’ll want to use different keys.

      Only use the keys generated from your Oracle Cloud account. Avoid using keys generated using external tools.

    - **Administrator User Name**: The default is `omceadmin`. You can change this if you wish.

      The name must have no fewer than eight characters and cannot exceed 128 ASCII characters. You cannot use a tab, brackets, parentheses or the characters , < > & # | ?.

    - **Administrator Password**: Enter a password for the administrator and re-enter it to confirm.

      This password is used across all suite components for their respective administrative users, for example, the WLS admin user. The password must have no fewer than eight characters and cannot exceed 30 characters. There must be at least one lower case letter, one upper case letter, one number, and one special character, either _ or #, for example, `Ach1z0#d`.

- **Hourly API Request Capacity**: The number of requests per hour you want your OMCe environment to handle.

  OMCe metering/billing will be based on this number, regardless of actual usage. In other words, if this value is set to 2500 requests per hour, then OMCe will bill 2500 * OMCe price per hour (Say 0.0021) = $5.25 per hour. This will continue regardless whether there is any actual traffic/requests through OMCe.

  > **Note:**
  >
  > Carefully review this number and do not accept default value. 2500 is considered high volume.

  For Dev/Test/Production environments, this is an average value of the requests per hour. Take the expected number of requests for the entire month, divide by 744, then enter this value.

  For trials, you can enter a relatively low number, like 10 or 50 requests per hour.

  Note that OMCe will continue to operate normally regardless of whether your actual usage is higher or lower than the value specified.

- **Install Sample**: Select to indicate that you intend to install the Bots samples from OTN.
- **Region**: Defines the data center within your data region where the OMCe instance will be provisioned. Select the data center location based on your actual location.

  > **Note:**
  >
  > This is a mandatory field. You must specify a region. If you don’t, OMCe components may be provisioned to different data centers which would result in provisioning failure.
  >
  > The region you select must be the same one that you specified for the database service.
  >
  > Do not use the default `No Preference`.

  For a mapping between data center code (e.g. `uscom-central-1`) and the actual location, see Selecting a Replication Policy for Your Service Instance.

- **Production Level of POD configuration**: Check if you are going to use Bots in a production environment.

10. For Storage Container Configuration:

    - **Cloud Storage Service URL**: This is the URL pointing to the Cloud Storage Service Container that will be used for OMCe. See Set Up the OMCe Service.

> **Note:**
>
> The URL format must be correct or stack creation will fail. It has the form `https://foo.storage.oraclecloud.com/MyService-bar/MyContainer`.

- **Cloud Storage User Name**: Enter the name of the Storage Cloud Service admin user you created earlier. This user should only have the Storage Cloud Service admin role, and nothing else.
- **Cloud Storage Password**: Enter the password for the Cloud Storage admin user.

11. Before you proceed, carefully review the requirement for the schema prefix which must be in all uppercase. Most important, be sure to use a different prefix for each OMCe environment/instance you create. If you are re-running an OMCe instance creation that previously failed, you must use a different schema prefix to avoid unexpected naming conflicts.
12. When you have finished entering the details, click **Next**.
13. Click **Confirm**.

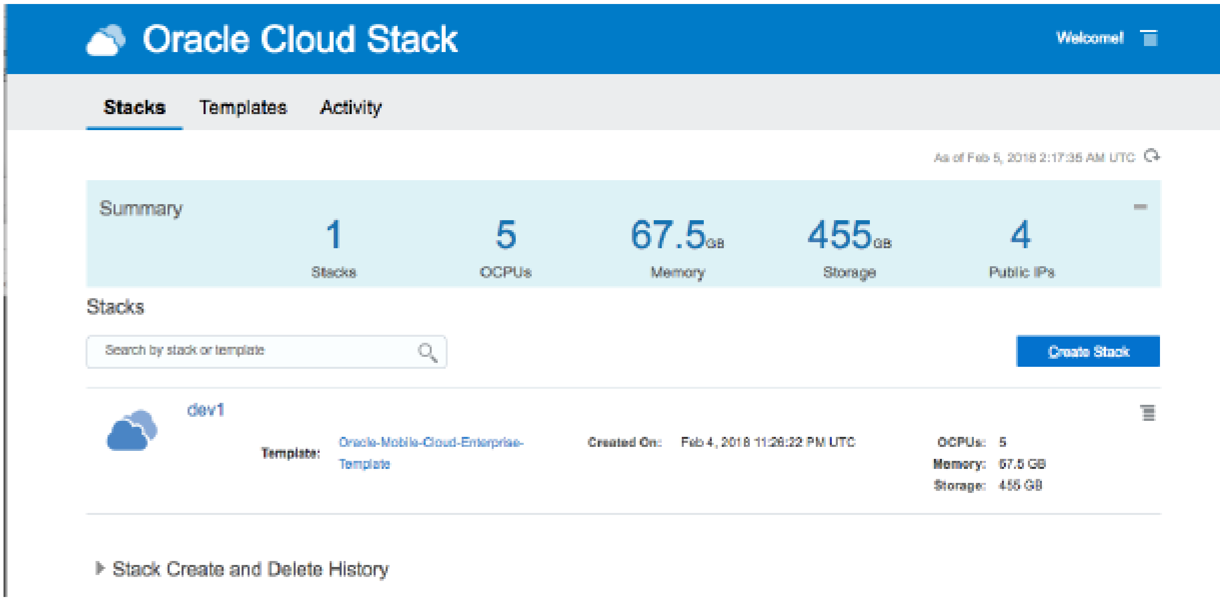
OMCe stack creation takes about 1-2 hours, and creates a number of resources. You can follow progress in the status pane which displays a list of the instances as they are created. Click the name of the service to find more information.

When the stack creation is complete, you’ll get a confirmation email.

Examine Your OMCe Environment

After you can have created your OMCe stack and received the confirmation email, you can have a look at the OMCe environment.

1. Log into MyService dashboard.
2. Click **Mobile**, then click **Open Service Console**. This opens the PSM (PaaS Service Manager) UI, which looks like this:



As of Feb 5, 2018 2:17:35 AM UTC

Summary

1	5	67.5GB	455GB	4
Stacks	OCPUs	Memory	Storage	Public IPs

Stacks

dev1

Template: Oracle-Mobile-Cloud-Enterprise-Template

Created On: Feb 4, 2018 11:26:22 PM UTC

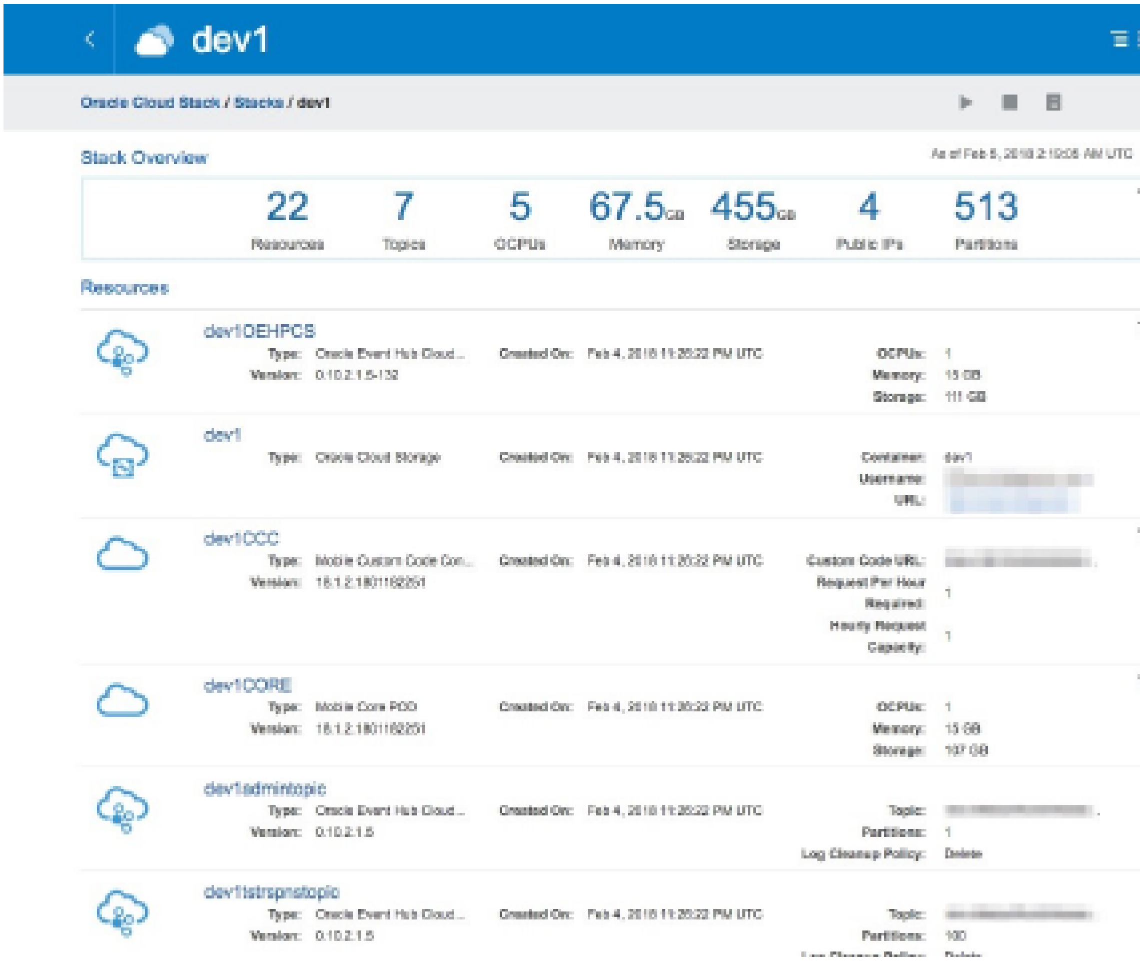
OCPUs: 5
Memory: 67.5 GB
Storage: 455 GB

This is where you can stop and re-start your OMCe instance in the future.

3. Click the name of the stack to drill into it and take a look at all the resources provisioned for OMCe.

   These resources include combinations of JCS, ACCS, Big Data CS, and Event Hub CS, and so on, all of which were provisioned for you behind the scenes.

   Your starting point is the resource called <*OMCe Name*>CORE. This is the portal through which you can enter OMCe.



dev1

Oracle Cloud Stack / Stacks / dev1

Stack Overview

As of Feb 5, 2018 2:10:06 AM UTC

22	7	5	67.5GB	455GB	4	513
Resources	Topics	OCPUs	Memory	Storage	Public IPs	Partitions

Resources

dev1OEHPCS
Type: Oracle Event Hub Cloud ...
Version: 0.10.2.1.5-132
Created On: Feb 4, 2018 11:20:22 PM UTC
OCPUs: 1
Memory: 15 GB
Storage: 111 GB

dev1
Type: Oracle Cloud Storage
Created On: Feb 4, 2018 11:20:22 PM UTC
Container: dev1
Username: [redacted]
URL: [redacted]

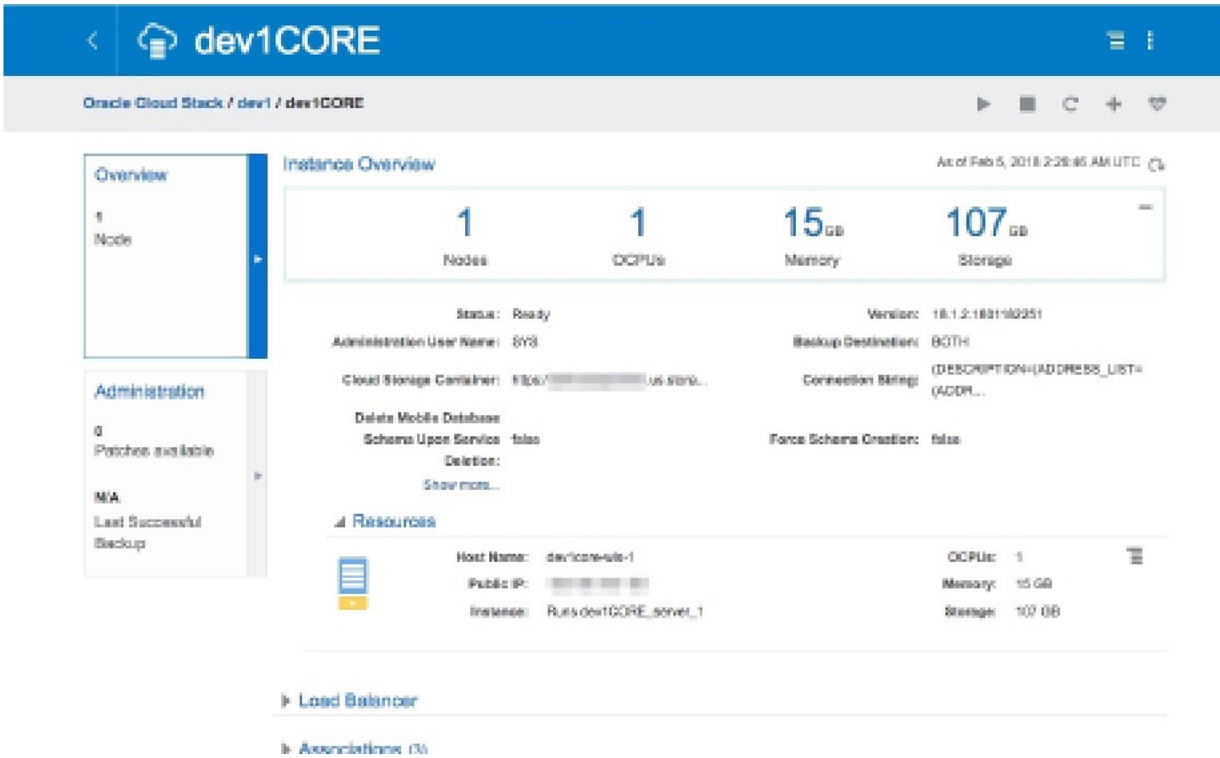
dev1CCC
Type: Mobile Custom Code Con...
Version: 18.1.2.1801162251
Created On: Feb 4, 2018 11:20:22 PM UTC
Custom Code URL: [redacted]
Request Per Hour Required: 1
Hourly Request Capacity: 1

dev1CORE
Type: Mobile Core POD
Version: 18.1.2.1801162251
Created On: Feb 4, 2018 11:20:22 PM UTC
OCPUs: 1
Memory: 15 GB
Storage: 107 GB

dev1admintopic
Type: Oracle Event Hub Cloud ...
Version: 0.10.2.1.5
Created On: Feb 4, 2018 11:20:22 PM UTC
Topic: [redacted]
Partitions: 1
Log Cleanup Policy: Delete

dev1lstrspnstopic
Type: Oracle Event Hub Cloud ...
Version: 0.10.2.1.5
Created On: Feb 4, 2018 11:20:22 PM UTC
Topic: [redacted]
Partitions: 100
Log Cleanup Policy: Delete

4. Click the <*OMCe Name*>**CORE** resource and examine the CORE service details:



dev1CORE

Oracle Cloud Stack / dev1 / dev1CORE

Overview

1
Node

Administration

0
Patches available

N/A
Last Successful Backup

Instance Overview

As of Feb 5, 2018 2:28:46 AM UTC

1 Nodes

1 OCPUs

15 GB Memory

107 GB Storage

Status: Ready

Administration User Name: SYS

Cloud Storage Container: https://[redacted].us.stora...

Delete Mobile Database Schema Upon Service Deletion: false

Show more...

Version: 18.1.2-1801182251

Backup Destination: BOTH

Connection String: (DESCRIPTION=(ADDRESS_LIST=(ADDR...

Force Schema Creation: false

Resources

Host Name: dev1core-wls-1

Public IP: [redacted]

Instance: Runs dev1CORE_server_1

OCPUs: 1

Memory: 15 GB

Storage: 107 GB

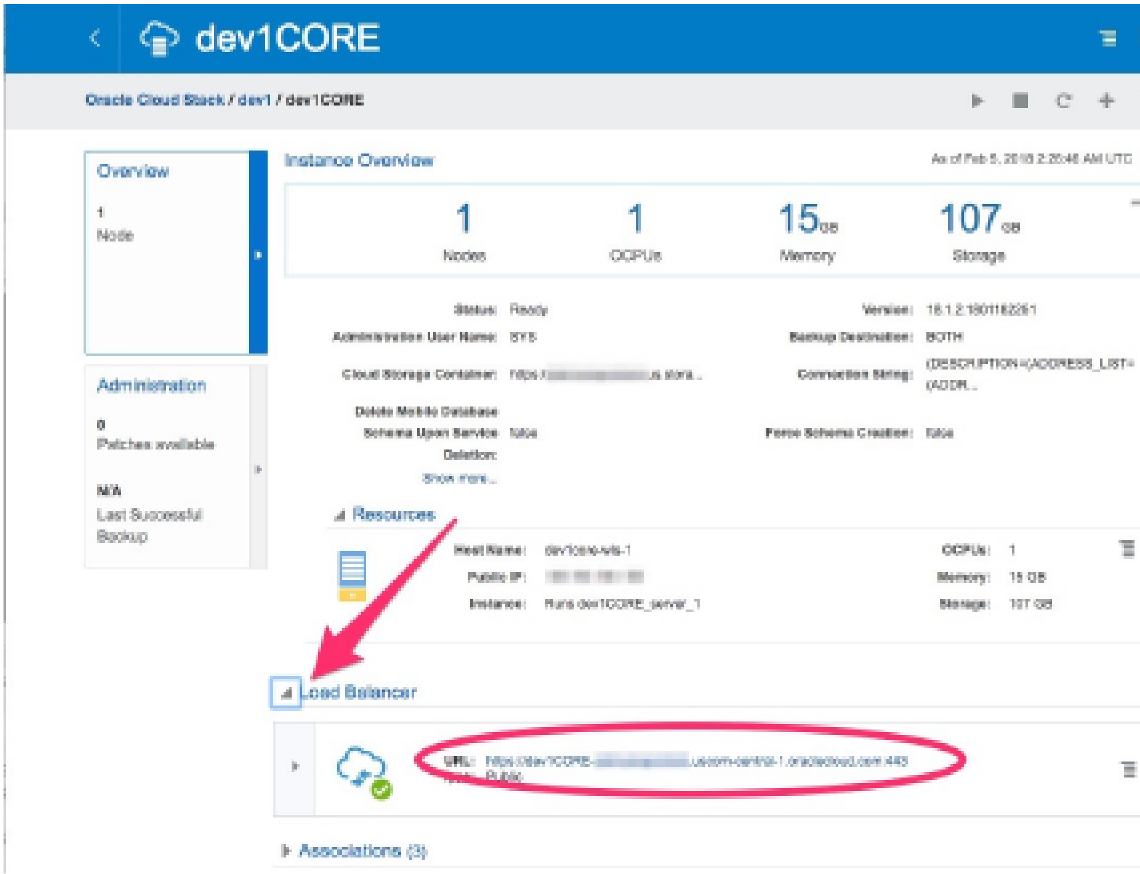
Load Balancer

Associations (3)

Log In to the OMCe UI

This is where you find the URL to log in to the OMCe UI.

1. Navigate to the <*OMCe Name*>CORE Instance Overview page.



dev1CORE

Oracle Cloud Stack / dev1 / dev1CORE

Instance Overview

As of Feb 5, 2018 2:20:46 AM UTC

1	1	15 GB	107 GB
Nodes	OCPUs	Memory	Storage

Status: Ready
Version: 18.1.2.180118281
Administration User Name: SYS
Backup Destination: BOTH
Cloud Storage Container: https://[redacted]us.stora...
Connection String: (DESCRIPTION=(ADDRESS_LIST=(ADDR...
Delete Mobile Database Schema Upon Service Deletion: false
Force Schema Creation: false
Show more...

Resources

Host Name: dev1core-wls-1
OCPUs: 1
Public IP: [redacted]
Memory: 15 GB
Instance: Runs dev1CORE_server_1
Storage: 107 GB

Load Balancer

URL: https://dev1CORE-[redacted].uscom-central-1.oraclecloud.com:443
Public

Associations (3)

2. Expand **Load Balancer**, and note the URL.

   To get to the OMCe home page, you need the URL `<Load Balancer URL>/mobileui/`.

3. Copy and paste this URL into a new browser window.

   You should now see OMCe UI, which you should bookmark. At this point, you’ll probably want to refer to *Developing Applications with Oracle Mobile Cloud, Enterprise* to get started.

Use IDCS to Manage Users and Roles

You can connect to IDCS from the OMCe Service console.

When you create the OMCe stack, an application for the OMCe core is created in Oracle Identity Cloud Service (IDCS).

To log in to the IDCS app:

1. On the Service Overview page of the Mobile Core POD for your OMCe service, expand **Show more**.
2. Click the link next to **IDCS App ID**. An instance of the IDCS service opens.
3. In the Identity Cloud Service console, click **Applications** to display a list of applications associated with the services in the OMCe suite.
4. Identify the OMCe core application, which has MobileCorePOD next to the name. Click the link for it.

The IDCS application for the OMCe application has the following tabs:

- **Details** - Displays high-level information about the app, it’s name and description, the icon for the app, the application URL, and whether or not the app is visible on the My Apps page.
- **Configuration** - Displays configuration information about your app.
- **Application Roles** - Displays the list of team member roles for your app. You can also import or export roles, assign or revoke users and groups to a role from this page. See Assign Roles.
- **Groups** - Displays the groups that are assigned to this app. See Create Groups.
- **Users** - Displays the users that are assigned to this app. See Add Users.

Add Users

Add OMCe team members by creating user accounts with Oracle Identity Cloud Service (IDCS). You can also use IDCS to provision mobile users of your apps and bots.

When you create an OMCe stack, an application is created in your identity domain for each service (OMCe core, Analytics, and Bots), and displayed on the IDCS Applications page. To enable a user to access a service, you must add the user to the associated application. For example, if a user needs to access OMCe core, Analytics, and Bots, then you must add the user to all three applications.

1. Log in to Oracle Identity Cloud Service (IDCS).

   You can do this from the Service Overview page of your OMCe instance, or you can use the URL for Oracle Identity Cloud Service which was provided in the Welcome email you received when you activated the OMCe account. In this case, use your OMCe credentials.

2. Click **Users**.
3. Click **Add**.
4. Enter the first name and last name of the user in the corresponding fields. Proceed to the next step to have the user log in with a user name or skip to Step 6 to have the user log in with an email address.
5. If the user is going to log in with a user name, enter the user name in the **User Name** field and enter the user’s email address in the **Email** field.

   Be sure to clear the **Use the email address as the user name** option, which makes the user name the same as the user’s email address.

6. If the user is going to log in using an email address, make sure the **Use the email address as the user name** option is checked and enter the email address for the user account in the **User Name/Email** field.

   Don’t use this option if the user name is currently in use or had been in use in the current domain. For existing (or previously existing) user names, a unique value will automatically be added to the email address to create a unique value for the user name.

7. Click **Next** if you want to assign the user to a group or click **Finish**.

   To assign a group, just select the groups that you want to assign to this user account and click **Finish**.

The Details page is displayed for the user. An invitation email is sent to the user containing log in credentials to OMCe.

8. Assign the user to an OMCe application by clicking the **Access** tab on the Details page.
9. Search for the application to assign the user to and click **Assign**.
10. Repeat the previous step for each service that you want to assign to the user.

    If you have many users to assign, a faster method is to use groups, described in Create Groups.

After you've added users, you can create groups and assign a user to multiple groups. You or a service administrator can assign roles.

Batch Import Users

If you have a large number of users to add, you can batch import user accounts using a comma-separated values (CSV) file.

You can access the Bulk Loading Users and Groups Using CSV Files tutorial to see how to import user accounts into IDCS.

Before you can import user accounts, first create a CSV file that is properly formatted for the import process. To create and prepare a file for import, follow these steps.

1. Use these sample files as a starting point.
2. Extract the compressed file and then open the `Users.csv` file.
3. Review and then delete any demo data in the `Users.csv` file.

   > **Tip:**
   >
   > To familiarize yourself with the import process, consider importing just the demo data. You can then delete the unwanted demo data from IDCS before you begin importing live data.

4. Create an import file using the `Users.csv` file. The `Users.csv` file is a simple text file in a tabular format (rows and columns). The first row in the file defines the columns (fields) in your table. At a minimum, the file must have these exact column headings.

   - User ID
   - Last Name
   - First Name
   - Work Email

   > **Tip:**
   >
   > As a best practice, ensure that the fields in these columns are unique.

   For each account, you create a new row (line) and enter data into each column (field). Each row equals one record.

> **Important:**
>
> The names of the users that you want to import into IDCS must contain at least three characters. The names of the groups that you want to import into IDCS must contain at least five characters.
>
> The telephone numbers of the users that you want to import must meet the requirements of the RFC 3966 specification.

To create a CSV file, you can use a standard spreadsheet application, such as Microsoft Excel or Google Sheets, or you can use a text editor, such as Notepad or TextPad.

5. Save your file in a CSV format.

> **Important:**
>
> If you do not save the file in a CSV format with UTF-8 encoding, the import fails.

To import user accounts:

1. In the Identity Cloud Service console, click **Users**.
2. Click **Import**.
3. In the **Import Users** dialog box, click **Browse** to locate and select the CSV file that contains the user accounts to import.

> **Note:**
>
> Click **Download sample file** in the dialog box to download a sample file.

4. Verify that the path and name of the CSV file that you selected appear in the **Select a file to import** field.
5. Click **Import**.

   If a user account is missing a required value, such as the user's first name, last name, or user name, then IDCS can't import it. If IDCS can't import a user account, then it evaluates the next account in the CSV file.
6. After IDCS evaluates all user accounts, review the job results.
   - If the job *can* be processed immediately, then a dialog box appears with the **Job ID** link for your import job. Click the link and review the details that appear on the **Jobs** page.
   - If the job *cannot* be processed immediately, then a message appears with a **Schedule ID** in it. Copy that **Schedule ID**, and use it to search for the job on the **Jobs** page. The job will appear when processing completes. Go to Step 7.

> **Tip:**
>
> IDCS assigns a job ID to each file that's imported or exported, for auditing purposes.

7. On the **Jobs** page, locate the job that you want to view, and then click **View Details**.

   A table displays the first names, last names, email addresses, user names, and statuses of the user accounts that you imported into IDCS.

8. Review the details that appear on the **Jobs** page.

   This page shows how many accounts you imported, how many accounts imported successfully, and how many accounts can't be imported because of a system error.

9. To make it easier to review and correct errors, you can export those errors to a CSV file on your local machine, In the Identity Cloud Service console, click **Jobs**.
10. Locate the specific job for which you want to export errors.

    You can only export errors for a job with a **Completed with Errors** status. Jobs with a **Failed** status do not have errors to export.

11. Click **View Details**.
12. Click **Export Errors**.

    A comma-separated value (CSV) file downloads to your local machine. The CSV file contains a record for each error that includes the error type and the error description.

Create Groups

Create groups to grant access to OMCe services to multiple users.

Use the IDCS console to create groups and assign users.

1. Log in to Oracle Identity Cloud Service.
2. Click **Groups**.
3. Click **Add**.
4. Enter a name and a brief description of the group.
5. Click **Next**.
6. Select each user that you want to assign to the group and click **Finish**. (Use the Search field to locate a specific individual.)

Next, assign roles to the group to grant access to OMCe services.

Assign Roles

Use roles to grant permissions and capabilities to the users in your OMCe service. Roles are managed from Oracle Identity Cloud Service (IDCS).

The predefined OMCe Team Member Roles are used to grant permissions and capabilities to the team members in your OMCe service.

Mobile user roles can be assigned in the same way as team members, but mobile user roles are not predefined. Use OMCe to create roles for your mobile users, as described in Mobile Users and Roles in *Developing Applications with Oracle Mobile Cloud, Enterprise*.

A service administrator can also assign OMCe roles to existing users. The identity domain administrator role is required to create new users and groups.

1. Log in to the IDCS console.
2. Click the **Applications** tab.
3. Locate and click the IDCS application for your OMCe service.
4. Click the **Application Roles** tab.
5. Open the role menu (☰) for the role you want to assign and select **Assign Users** or **Assign Group**.
6. Select the users or groups that you want to assign the role to and click **Assign**.

You can also use the role menu to revoke a role from a team member or group.

OMCe Team Member Roles

OMCe team member roles are predefined and can’t be created or customized. These roles determine which team members can develop backends, custom APIs, and use built-in services, such as Notification, Storage, and more. You can give team members wide access to features and user information or restrict them to a small set of permissions.

> **Note:**
>
> Team members must be assigned at least one of the roles in the table below to access the OMCe UI.

The common roles performed by team members in an OMCe instance are Administrator, Developer, Deployer, and Business User.

Role Name	Team Member Role
Service Administrators	Performs administrative actions, including managing and overseeing the OMCe instance. This role is used with OMCe core, Analytics, and Bots. A specific type of administrator is the Entitlement Administrator. This role is assigned to the administrator when OMCe is first ordered, and it is the role used by the service administrator to create and manage OMCe.
OMCe Developer	Performs development and testing tasks, accesses MAX from within OMCe. This role is used with OMCe core and Analytics.
Deployer	Performs configuration tasks as well as artifact and metadata export and import tasks. Publishes apps from the MAX production environment. This role is used with OMCe core.
Business User	Creates apps only with Mobile Application Accelerator (MAX). **Note:** This role limits the user to the MAX UI. Do not assign this role to app developers. The ServiceDeveloper role grants app developers access to MAX from within OMCe. This role is used with OMCe core.

You might see some other predefined roles in the list for OMCe in Oracle Cloud, but they are for internal use and shouldn't be assigned to any team members.

View and Edit Users

From the IDCS **Users** page, you can view and edit profile information for all user accounts and all the groups to which a user is assigned.

1. Log in to the IDCS console.
2. For details about a user, click the **Users** tab.
   a. Enter all or part of a user name in the **Search** field to locate a specific user. You can also filter your search by selecting a user status: `Active`, `Inactive`, or `Locked`.
   b. Click a user name to see and edit details for that user, for example:
      - The user's primary email address to which notifications are sent.
      - The user's password recovery email address, to which notifications are sent if the user forgets the original password.
      - The user's instant messaging address.

- Whether or not the user has a federated single sign-on (SSO) account. With a federated account, the user can interact with IDCS through an identity provider by using a web site that's external to IDCS.
- Work information, including the user's country, time zone, and preferred language.

c. Click the **Groups** tab to see which groups are assigned to the user.

   You can also revoke group privileges by selecting a group and choosing **Revoke** from the menu.

d. Click the **Access** tab to see a list of any applications assigned to the user.

3. For details about a group, from the **Home** page, click **Groups**. All the groups that have been added are listed.
   a. Click a group to see its description.
   b. Click the **Users** tab to see who is assigned to the group.
   c. Click the **Access** tab to see all the applications assigned to the group.

Reset User Passwords

As the service administrator, you need to ensure that OMCe user information is kept up to date. You can do this through Oracle Identity Cloud Service (IDCS). When you request a password change for a user account, the user receives a notification to provide a new password.

You can reset the password for a single account or multiple accounts in an identity domain.

> **Note:**
>
> You can't reset the password for deactivated user accounts. However, you can reactivate those accounts.

1. Log in to the IDCS console and click **Users**.
2. Select the user accounts for which you want to reset the password.
3. Click **More** and perform of the following actions:
   - For a single or multiple user accounts, select **Reset Password**, then click **Reset Password** in the dialog.
   - For resetting the passwords for all user accounts, select **Reset All Passwords**, then click **Reset Passwords** in the dialog.

5 Managing an OMCe Stack

After you've created an Oracle Mobile Cloud Enterprise (OMCe) service, you'll need to monitor various aspects of it. You can easily do this from the Oracle Cloud Stack Manager. You'll be able to get a summary of your OMCe stack, view performance metrics, storage, and database information for the stack and for each resource used in the stack and any services associated with those resources. From the PaaS Service Manager Services page, you can view details for and monitor the operations for the individual components of the OMCe stack.

A cloud stack is a collection of related cloud services that you've provisioned. A single stack represents a complete OMCe environment (OMCe core, custom code components, Analytics, and Bots, which are the individual elements of the stack).

Sign in to Oracle Cloud Stack Manager, to see a summary of information for your OMCe service. Click a tab to get service and deployment, operation, and security information:

- Click the **Stacks** tab to create a stack and to see a summary of the resources currently being used and general service information for an existing stack:
  - **Summary**. For all existing stacks in this identity domain, you can see:
    - How many configured deployments you have.
    - The total number of Oracle CPUs (OCPUs) that are allocated across all your deployments.
    - How much memory is allocated across all your deployments.
    - The total amount of Oracle Compute Cloud Service storage is allocated across all your deployments.
    - The number of public IP addresses are allocated across all your deployments.
    - The list of existing stacks in this domain. For each stack, there is a link to the stack template, the date the stack was created, the OCPU, memory, and storage allocation for that stack, and the Manage this stack menu.

      Select the Show only failed attempts option to see only a list of stack creation and deletion jobs that failed. Click a job to see more information about it.
  - **Stack Creation and Delete History**. Expand this section to see a history of when stacks were created or deleted over the last 24 hours, the last seven days, or the last month. You can also choose to view all failed creation and deletion jobs.
- Click the **Templates** tab to view the details of the stack template. From the Templates tab, you can:
  - View the topology of a template.
  - Import or export a template.
  - Update or delete a template.

> **Note:**
>
> You should never alter an OMCe template. When you provision a service, use the supplied OMCe template.

- Click the **Activity** tab to view the operations for all of your OMCe stacks in the current identity domain. Use the Search Activity Log to filter your results by start time, range of time, service name, service type, the type of operation, and operation status.
  - See the time range for the listed activity results.
  - For each operation, view the type of operation, the service instance, the operation status, the start and end time of the operation, and who initiated that operation.

If you have questions, need assistance, or want to discuss your experiences with other users in the Oracle Cloud community, click the Contact Us (?) .

In addition, you can also manage your OMCe stack using the Platform-as-a-Service Manager Command-Line Interface (PSM CLI) or directly with REST APIs. See Managing OMCe with the PaaS CLI.

View Your OMCe Stack

As the service administrator, you can examine information about your OMCe service instance using Oracle Cloud Stack Manager.

To view information about the stack:

1. Sign in to the Oracle Cloud Stack Manager.

   You'll need to provide the access information that you received in the Welcome email (the identity domain name and user name and password).

   You can see an overall summary of the OMCe stack, which lists the number of stacks, the of Oracle CPUs (OCPUs) in use, the amount of memory and storage used, and the number of public IPs used for the entire stack. In the Stack list, you can see the same information for each stack you've created.

2. Click on a stack icon or name to see its details.

   A stack overview is displays displaying resource, OCPU, memory, storage, and public IP allocated for the selected stack.

   For each resource, you can see its name, type, version, creation date, and the number of OCPUs, memory, and block storage that are allocated.

   Click the Action menu (☰) for a resource to start, stop, or restart the resource.

   You start or stop all the resources listed by click the **Start** (●) or **Stop** (◉) icon. Click (📄) to view the stack template.

3. Click a resource name to see its details.

   You can see the latest information for the resource:

   - Service information, including status, compute shape, deployment type, version, and connect descriptor address.

- Resource details, including host name, public IP address, the instance the resource runs, the OCPU, memory, and storage allocation.
- Associated services, their name, service type, status, and the whether the resource is a member of an associated service or is hosted on the service.
- The number of nodes in the resource and the partitions used and total partitions available.
- The Administration panel shows the number of patches available. Click Administration to review patching information.

View Stack Activity

You can see if an operation on a stack has successfully completed or failed:

1. Sign in to Oracle Cloud Stack Manager.
2. Click the **Activity** tab.

   Use the filter field to search the stack activity log by start time, a specific time range, operation status, service name, service type, or operation.

Review the log table to see information for each operation in the stack, its service name and type, operation status, start and end times, and who initiated the operation.

Start and Stop an OMCe Stack

There are times when you need to stop OMCe resources within a stack. For instance, you'll need to stop it when you need to apply a patch or debug some issue. Use Oracle Cloud Stack Manager to start and stop a stack. Dependencies between the resources in the stack determine the order in which they stop or start. You can also start and stop a single resource in the stack.

How Do I Start or Stop a Stack with the Console?

Use the Stack Manager UI to start or stop a stack:

1. Sign in to Oracle Cloud Stack Manager.

   If you're viewing the details of a specific stack, you can also start or stop it from the **Stack Details** page.
2. Locate the stack you want to start or stop and open the **Manage this service** (☰) menu.
3. Select **Start** or **Stop**.

   Confirm your action in the dialog.

The stack's icon changes to indicate that it's in either the starting or stopping state. You can monitor the operation's progress from the **Activity** page.

Deleting an OMCe Stack

You can delete a stack when you no longer need it to free up resources. Deleting a stack terminates all the resources in that stack.

How Do I Delete an OMCe Stack with the Console?

1. Sign in to Oracle Cloud Stack Manager.
2. On the Stacks tab, locate the stack that you want to delete.
3. Open the **Manage this service** (☰) menu and select **Delete**.

   Click **OK** in the confirmation dialog.

Add a New SSH Public Key

Some Oracle Cloud resources are accessed through the Secure Shell (SSH) protocol. You provide a key when create such a resource. If you need to add more keys to a resource in your OMCe stack, you can add them with Oracle Cloud Stack Manager.

1. Sign in to Oracle Cloud Stack Manager and navigate to the Stacks page.
2. Click a stack icon or name to go to its Details page.
3. Click the **Manage this resource** menu (☰) for the resource you want to update and select **SSH Acess**.

   This menu item only appears for resources accessed by an SSH Public Key.
4. On the SSH Access page, locate the resource and click **Add New Key**.
5. Add a new key in one of the following ways:
   - Upload a new SSH Public Key from an existing file.
   - Select the Key value option and paste the value of the new public key in the text pane.
6. Click **Add New Key**.

Return to the Stacks page by clicking **Stacks** in the navigation breadcrumb on the page.

Add an Access Rule

Set access rules to control access from external clients or other Oracle Cloud resources to the components that make up a resource. Default access rules are created by default by Oracle Cloud when the resource is provisioned; however, you can create custom rules to control access.

1. Sign in to Oracle Cloud Stack Manager and navigate to the Stacks page.
2. Click a stack icon or name to go to the Details page.
3. Click the **Manage this resource** menu for the resource you want to update and select **Access Rules**.
4. On the Access Rules page, click **Create Rule**.
5. Provide the following information:
   - A name for the Access Rule.
   - (Optional) A description of the rule.

- Select a source. The source selections listed depend on the type and configuration of the resource.
- Select a destination. Like the Source field, the destination selections listed depend on the type and configuration of the resource.
- Enter the port or port range that the source uses to access the destination you selected.
- Select a protocol type, either TCP or UDP.

6. Click **Create**.

Return to the Stacks page by clicking **Stacks** in the navigation breadcrumb on the page.

6

Patching OMCe

This section describes how to apply a patch to the services comprising an OMCe stack, and roll back the patch as necessary.

You can patch an OMCe stack from the Service Details page of the Service Manager, or by using the Patch Tool which is a command line script. Patching occurs at the level of individual services, but because OMCe is a stack all services within it must be patched at the same time. You cannot patch just some of the services comprising an OMCe stack.

Download the Patch Tool

The Patch Tool is part of the Admin tools zip which you can get from the OMCe Downloads page on OTN at http://www.oracle.com/technetwork/topics/cloud/downloads/mobile-suite-3636471.html. Save the zip to a local location and unzip it.

You must have Python 3 installed on the same machine from where you will run the patch. You can get it from https://www.python.org/downloads/.

Note:

The OMCe Admin tools use the executable `python`. If the version of Python 3 you have has the `python3` executable you should either modify the `patch.sh/patch.cmd` scripts to use `python3`, or symlink `python` to the `python3` executable.

On OSX, if you get an error `SSL: CERTIFICATE_VERIFY_FAILED` when you run the tool it is probably because Python 3.6 on OSX has no certificates and can't validate any SSL connections. You can install the `certifi` package of certificates. See the ReadMe at `/Applications/Python\ 3.6/ReadMe.rtf` .

OMCe Service Instances

The OMCe stack is comprised of a number of services which you can see in the Stack Overview page for your OMCe stack. The name of the service has the stack name as the first part of the service name:

- {stack}CORE
- {stack}CCC
- {stack}BOTSCFG
- {stack}BOTSCON
- {stack}BOTSINT
- {stack}BOTSPIP
- {stack}BOTSMGM

- {stack}OMCE
- {stack}CXACFG
- {stack}CXAPOD
- {stack}1CXAANA
- {stack}1CXACOL

Typical Workflow for Patching an OMCe Stack

Consider the typical workflow for patching an OMCe stack, as described in the following table.

Task	Description	More Information
Learn about approved patches	View approved patches displayed on the Patching page periodically.	Viewing Patch Details
Perform optional prechecks	Learn about problems that would cause patching to fail, so you can address those problems before you try to apply a patch.	Performing a Patch Precheck
Apply a patch	Initiate a patching operation to update the service instances which comprise and OMCe stack.	Patching OMCe Services
Roll back a patch	Initiate a rollback operation to return a service to its previous patch level.	Rollback a Patch with the PaaS CLI

Configuring the Patch Script

If you just run the patch script, you will be prompted several times to enter information about your OMCe log in and IDCS instance. You can avoid this by using `patch.json`, which is provided as part of the OMCe Admin Tools. You may want to leave the value for `psmPassword` blank, in which case you will be prompted for the password when the script is run.

The default `patch.json` is:

```
{
  "psmUser": "<OMCe admin user> ",
  "psmPassword": "<password for OMCe admin user>",
  "psmUri" : "<URI for PSM Stack Manager>",
  "psmIdentityDomain" : "<IDCS domain>"
}
```

The URI for the PSM Stack Manager has the form `https://`*`region-prefix`*`.oraclecloud.com`, where the *`region-prefix`* is the URL prefix based on region. Typical values include:

- United States: `psm.us`

  For example: `https://psm.us.oraclecloud.com`

- Europe: `psm.europe`

  For example: `https://psm.europe.oraclecloud.com`
- Australia: `psm.aucom`

  For example: `https://psm.aucom.oraclecloud.com`

Viewing Patch Details

Use the Patching page to view information about the available patches, as well as your patch history.

1. From the Instance Overview page, use the Administration tile to view the number of available patches.
2. If any patches are available, click the **Administration** tile.
3. Click the **Patching** tab.
4. In the Available Patches section click the icon for a patch.

   An overlay displays with more details about the patch.

You can also use the patch script to check whether the stack is ready to be patched. Use:

```
./patch.sh -s {stackName} patch_precheck
```

The response will list the services and the stack is ready to patch when the status for all services is `Ready to apply patch`.

Before applying any patches, Oracle recommends that you perform a precheck. See Performing a Patch Precheck.

Performing a Patch Precheck

You can optionally perform a precheck operation before applying a patch in order to identify and remedy any issues.

The precheck operation helps detect potential patching problems such as:

- Disk space shortage
- Missing database connectivity
- Servers not running
- Storage access failure

The precheck operation does not check whether another administration task (backup, restoration, or scaling) is in progress, which prevents patching.

1. Access your service console.
2. Click the name of the service instance to which you want to apply a patch.
3. Click the **Administration** tile.
4. Click the **Patching** tab.
5. Beside the patch that you want to precheck, click **PSU** ☰ , and then select **Precheck**.

6. To determine whether the prechecks are completed, periodically click **Refresh** ⟳.

   The **Precheck results** link is displayed.

   - If prechecks pass, a green check mark icon is displayed.
   - If prechecks fail, a red exclamation mark icon is displayed.

7. Click **Precheck results**.
8. If there are precheck failures, address the issues, and then run **Precheck** again.

After addressing any precheck failures, you are ready to apply the patch. See Patching OMCe Services.

Upgrading from 17.4.5 to 18.1.1

If you have previously patched from 17.4.3 to 17.4.5, then before you patch to 18.1.1 there is an additional step you need to perform. You have to connect to the Core VM using ssh and run a script, and then run the patch.

> **Note:**
>
> This only applies to OMCe instances that were provisioned on release 17.4.3 and have been patched to 17.4.5. If you are patching directly from 17.4.3 to 18.1.1 you don’t need to run the pre-patch script.

How do I check whether my OMCe instance may be affected?

If you are unsure whether your OMCe instance was initially provisioned on 17.4.3 and patched to 17.4.5, you can check the patching and rollback history.

1. Navigate to the Patching page. Click Administration tab on the left, then select Patch.
2. Expand Patch and Rollback History. If you can see that the 17.4.5 patch has been applied then your OMCe instance was provisioned at 17.4.3 and you need to perform this step.

How to run the pre-patch script

1. Find the Core VM’s IP address.

   a. From OMCe Stack Overview, click the resource with the name `<stack>CORE`. See View Your OMCe Stack.

   b. In the Service Overview for the Mobile Core POD Details, expand Resources and note the Public IP, for example `192.0.2.1`.

2. Connect to the Core VM using ssh and change to the `oracle` user. You can log in as the default user, `opc`. The `opc` user has `sudo` privileges.

   a. Use:

   ```
   $ ssh -i <private_key_file> opc@<Core IP address>
   ```

   For example, `ssh -i my_private_rsa_keyfile opc@192.0.2.1`

   b. Switch to the `oracle` user using:

```
$ sudo su - oracle
```

3. Save the script below as `prepatch_omce_18_1_1.sh` in a convenient location, give `exec` permissions for current user, and run the script.
4. Check that the output looks correct. It should look similar to this:

```
Updating: /u01/data/domains/<domain-name>/init-info/domain-info.xml
Updating: /u01/data/domains/<domain-name>/init-info/config-path.properties
Updating: /u01/data/domains/<domain-name>/init-info/startscript-unsub.xml
```

   If it is, you can continue to patch to 18.1.1.

The pre-patch script

Save this as `prepatch_omce_18_1_1.sh` in a suitable location.

```
#!/bin/bash

_domain_configs="domain-info.xml config-path.properties startscript-unsub.xml"

_domain_init_info_dir=$(find "/u01/data/domains" -type d -name "init-info")

for f in ${_domain_configs}; do
  _fpath="${_domain_init_info_dir}/${f}"
  _backup="${_fpath}.bak"
  if [ -f "${_backup}" ]; then
    echo "Skipping ${f} because its backup already exists at ${_backup}"
    continue
  fi
  echo "Updating: ${_fpath}"
  sed -i.bak "s/17.4.3.0.0/17.4.5.0.0/g" "${_fpath}"
done
```

Patching OMCe Services

Do not apply a patch to service instances:

- If one or more of the service instance has any ongoing maintenance operations, such as backup, restoration or scaling
- If any database deployment associated with one or more of the service instances is undergoing a patching operation

Patch All OMCe Services

Oracle recommends that you use the patch script to patch all the services in an OMCe stack. Use the command:

```
./patch.sh -s {stackName} patch
```

If you accidentally exit the script before it has completed don't panic. The patch operations that are currently in progress will continue. Simply make the following call to check status a few times until you see no running activities on any of the service instances.

```
./patch.sh -s {stackName} status
```

Once you see that there are no running instances, re-run the patch command. The patching command will not modify service instances that have already been patched, and will resume patching where it is required.

Patch Individual OMCe Services

You can patch services one by one. There is a recommended order in which the OMCe components should be patched, and within the components, the order in with the services should be patched. Use this list to guide you through the order of applying patches.

> **Note:**
>
> Oracle recommends that you use the patch script to patch all the services in an OMCe stack in one go. This information is provided in case there is a problem with the script and you are advised to patch services one by one.

1. Core
   a. {stack}CORE
   b. {stack}CCC
2. Bots
   a. {stack}BOTSCFG
   b. {stack}BOTSCON
   c. {stack}BOTSINT
   d. {stack}BOTSPIP
   e. {stack}BOTSMGM
   f. {stack}OMCE
3. Analytics
   a. {stack}CXACFG
   b. {stack}CXAPOD
   c. {stack}1CXAANA
   d. {stack}1CXACOL

Applying a Patch Using the Patch Script

Patch an individual service using:

```
./patch.sh -t {stackType} -n {serviceName} patch
```

Applying a Patch from the Service Details Page

If you do not want to use the patch script you can use the Patching page of the Service Details page for the service you want to patch.

1. Navigate to the Patching page. Click Administration tab on the left, then select Patch.

2. Click ≡ to the right of the patch you want to apply and select **Patch**. The Patch Service dialog is displayed.
3. In the **Notes** field enter a description for this patching operation. You must enter a note. If you do not, an error message will be displayed and you will not be able to patch.
4. Click **Patch**.

   The patching operation begins, and the patches are applied in a rolling fashion, having minimal impact on the functioning of the service.

   - A message stating that a patch job has been submitted for the service is displayed in the upper left corner of the console.
   - The Administration tile displays a message about the version of patch being applied. It also shows the time of the backup that occurred before the patching operation started.
   - The ≡ menu for this patch and all other available patches are disabled.

   You can follow the progress of the patch by clicking the Overview tab. Use the ⟳ at the upper right to refresh the in-progress operation messages.

Upgrading Bots Samples After Patching

If you are patching from 17.4.3 to 17.4.5, you will have to recreate the sample of financial Bots. The new financial bot will work with the 17.4.5 samples container.

1. After patching your system, go to the Sample Bot container.
2. Select the Recreating Sample Bots tab and download `financialBot.json`.
3. In the patched environment, import the downloaded `financialBot.json`.

Troubleshoot Patch Issues

Sometimes patching goes wrong. When the patch has been unsuccessful rollback is automatically performed to restore the service instance or instances to their original state. But sometimes there are issues where you need to intervene to resolve issues so that you can re-attempt the patch.

Troubleshoot Using the Patch Script

The patch script writes logging information to a file called `patch.log` which is in the same directory as the patching script. `patch.log` contains information you will find useful when you are diagnosing the problem that has caused the patch failure.

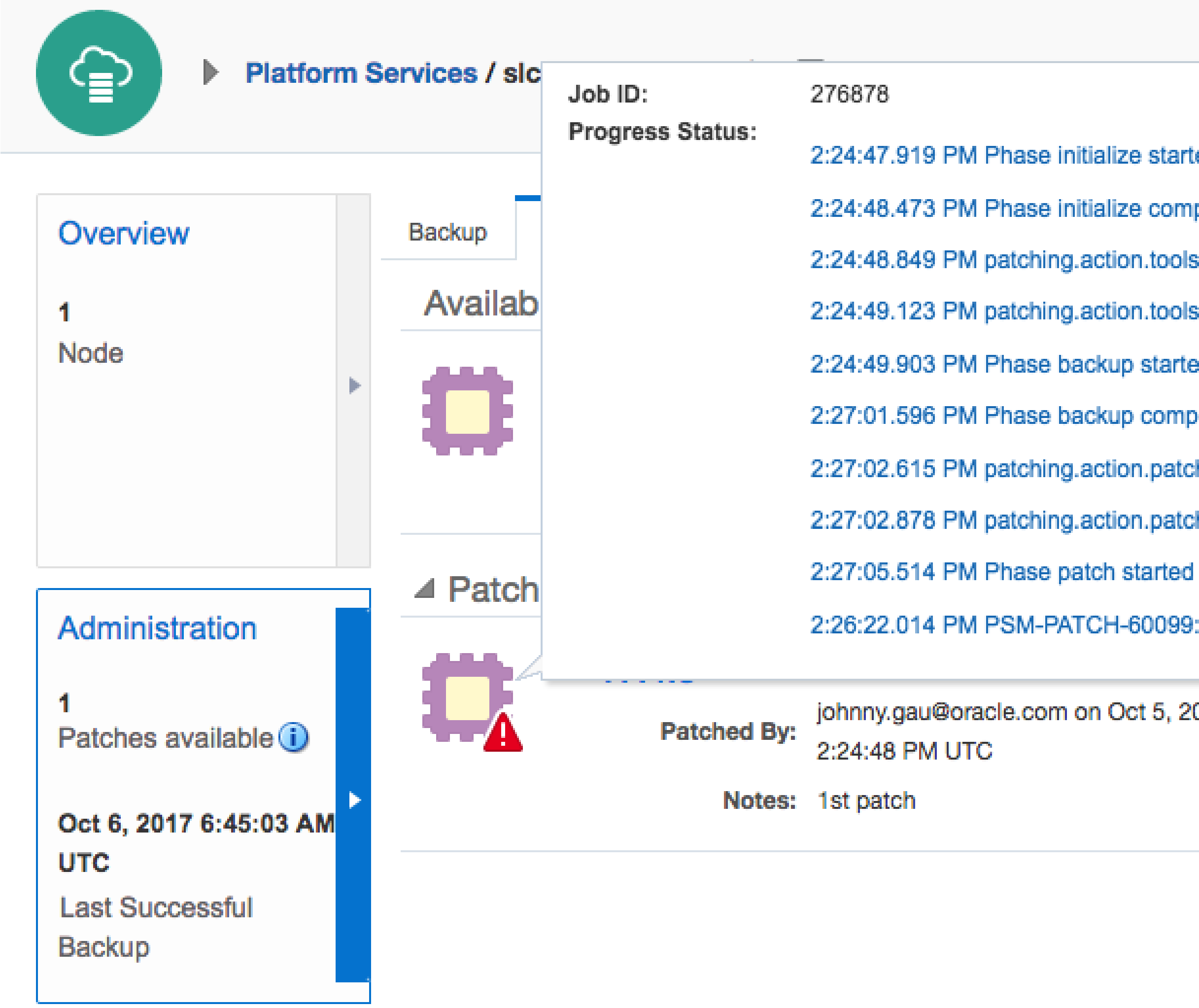
You can gather even more information by re-running the patch operation specifying a log level of "DEBUG". Use:

```
./patch.sh -t {serviceType} -n {serviceName} -ll DEBUG patch
```

Troubleshoot from the Patching Page

You can use information on the Patching Page of the Service Details page for the service to diagnose why a patch or a rollback operation has failed to complete successfully regardless of whether you ran the script from the patch script or from the Service Details page.

1. Navigate to the Patching page. Click Administration tab on the left, then select Patch. Failed operations have a red triangle on the icon.
2. Click the icon to show details.



Job ID: 276878

Progress Status:

2:24:47.919 PM Phase initialize starte
2:24:48.473 PM Phase initialize comp
2:24:48.849 PM patching.action.tools
2:24:49.123 PM patching.action.tools
2:24:49.903 PM Phase backup starte
2:27:01.596 PM Phase backup comp
2:27:02.615 PM patching.action.patc
2:27:02.878 PM patching.action.patc
2:27:05.514 PM Phase patch started
2:26:22.014 PM PSM-PATCH-60099:

Patched By: johnny.gau@oracle.com on Oct 5, 20
2:24:48 PM UTC

Notes: 1st patch

Keep a note of the Job ID in case you have to contact support for assistance.

Rolling Back a Patch

Use the patch script or the Patching page to roll back a patch that you previously applied.

If you have to roll back a patch, you do this by rolling back individual services so that they are all rolled back to the same version.

> **Note:**
>
> You must ensure that all services are patched or rolled back to the same version for OMCe to function correctly. This means that if one of the services fails to upgrade, all services that have been patched must be rolled back.

Rollback Using the Patch Script

Use the following commands to rollback an OMCe stack or an individual stack.

Check stack is ready to have rollback applied:

```
./patch.sh -s {stackName} rollback_precheck
```

Check service is ready to have rollback applied:

```
./patch.sh -t {serviceType} -n {serviceName} rollback_precheck
```

Rollback the OMCe stack:

```
./patch.sh -s {stackName} rollback
```

Rollback an individual service:

```
./patch.sh -t {serviceType} -n {serviceName} rollback
```

Rollback From the Service Details Page

Use the Patching page of the Service Details to rollback one service at a time. Rollback the services in reverse order that the patches were applied. See Patch Individual OMCe Services.

1. Navigate to the Patching page. Click Administration tab on the left, then select Patch.

2. Click ☰ to the right of the patch you want to apply and select **Rollback**.

7 Manage OMCe Services

As the service administrator, you use this page monitor the activity for individual services in the OMCe stack.

1. Sign in to Oracle Cloud Stack Manager.
2. Click the Stack menu and select the OMCe service you want to examine.
3. Click the **Services** tab to see a summary of the resources currently being used and general service information:
   - How many configured deployments you have.
   - The total number of Oracle CPUs that are allocated across all your deployments.
   - How much memory is allocated across all your deployments.
   - The total amount of Oracle Compute Cloud Service storage is allocated across all your deployments.
   - The number of public IP addresses are allocated across all your deployments.
   - Service Creation and Delete History. Expand this section to see a history of when deployments were created and deleted over the last 24 hours, the last seven days, or the last month.

     Expand **Details** to see more information about each event.

   From the Services tab, you can start, stop, restart, or delete a service. If applicable to the service, you can also modify access rules or add a new SSH Public Key.
4. Click the **Activity** tab to view the operations for all of your OMCe deployments in the current domain. Use the Search Activity Log to filter your results by start time, range of time, service name, service type, the type of operation, and operation status.
   - See the time range for the listed activity results.
   - For each operation, view the type of operation, the service instance, the operation status, the start and end time of the operation, and who initiated that operation.
5. Click the **SSH Access** tab to view and add SSH public keys to the OMCe deployments within the current domain. Use the Search filters to search based on service name or service type. Note that this tab appears only if SSH access is applicable to the service.
   - For each service deployment, see the name and type of the deployment and when it was last updated.
   - For each active deployment, add a new SSH public key:

     When you click **Add New Key**, you can either upload a new key or delete the key value displayed and paste a new public key in the text field. If you paste a new key value in the field, be sure that the value doesn’t contain line breaks or ends in a line break.

If you have questions, need assistance, or want to discuss your experiences with other users in the Oracle Cloud community, click the Contact Us (?) .

Other things you can do on the My Services page is use the Oracle Cloud Services menu (☰) to access the Oracle Cloud Stack, use the navigation menu (▤) to access other cloud services in the current domain, click **Welcome!** to go to the OMCe service Welcome page.

In addition, you can also manage your OMCe service using the Platform-as-a-Service Manager Command-Line Interface (PSM CLI) or directly with REST APIs. See Managing OMCe with the PaaS CLI.

Archive and Purge Bots Data

Large amounts of Bots conversational data is written to the database. Over time, this can grow to a level that causes a database exception unless you take action to archive and purge this data. The specific table that the conversational history is written to is `BOTS_CONVERSATION_HISTORY_`. See Backing Up and Restoring Databases on Database Cloud Service in *Administering Oracle Database Cloud Service*.

8

Troubleshoot Your OMCe Service

Here’s where you’ll find information to help you resolve any problems you come across.

It’s likely that at some point when you’re creating and maintaining your OMCe service, you’ll run into issues or just have questions about the proper way to proceed. Review the information presented here to see if solutions to your issues or questions are addressed and where to go to get help if they aren’t.

Where Do I Get Help?

You can get assistance if you need help resolving an issue with your OMCe service by phone or using chat. On the Oracle Cloud My Service dashboard, or from the Stack console or Service console, click ? to see the phone number and the Chat link.

Using Diagnostic Tools

If you contact Oracle Support they may ask you to download and run one of the diagnostic tools:

- Mobile Suite Doctor. See Using the OMCe Mobile Suite Doctor.
- Mobile Core Doctor. See Using the OMCe Mobile Core Doctor.

What Information Do I Need?

Before you contact Oracle Support, make sure you have your Customer Support Identifier (CSI) number, which is in the Welcome email. In addition, you should also have:

- The identity domain name which you can find on the Details tab of the Mobile Core POD IDCS app. See Use IDCS to Manage Users and Roles.
- The user name for OMCe.
- The OMCe stack name.
- Status information from My Services including the following. See Troubleshoot Mobile Suite Provisioning
  - Content of the Stack Create and Delete History.
  - Summary messages from the details page you see when you click the Status error link.

For more information about getting log messages when Oracle Support ask for them, see Getting Log Messages.

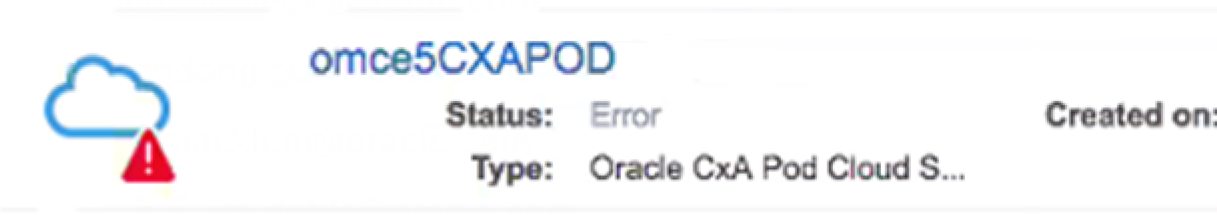
Troubleshoot Mobile Suite Provisioning

If stack creation fails, or one or more services fail when you are provisioning OMCe, use the Create and Delete histories available from the dashboard to try to identify where the problem is.

How to Recognise a Problem

The process of creating an OMCe stack can take a few hours. During this time you can see the status of different resources in the Stack Overview window:

- When a stack is being created, there is an hourglass icon and the status is a clickable link **Creating Stack** which you can use to get further information.
- When service creation fails, the relevant entry has an error icon and the status is **Error**. You can click on the status link to see more details in the Stack Create and Delete History pane.



omce5CXAPOD

Status: Error

Type: Oracle CxA Pod Cloud S...

Created on:

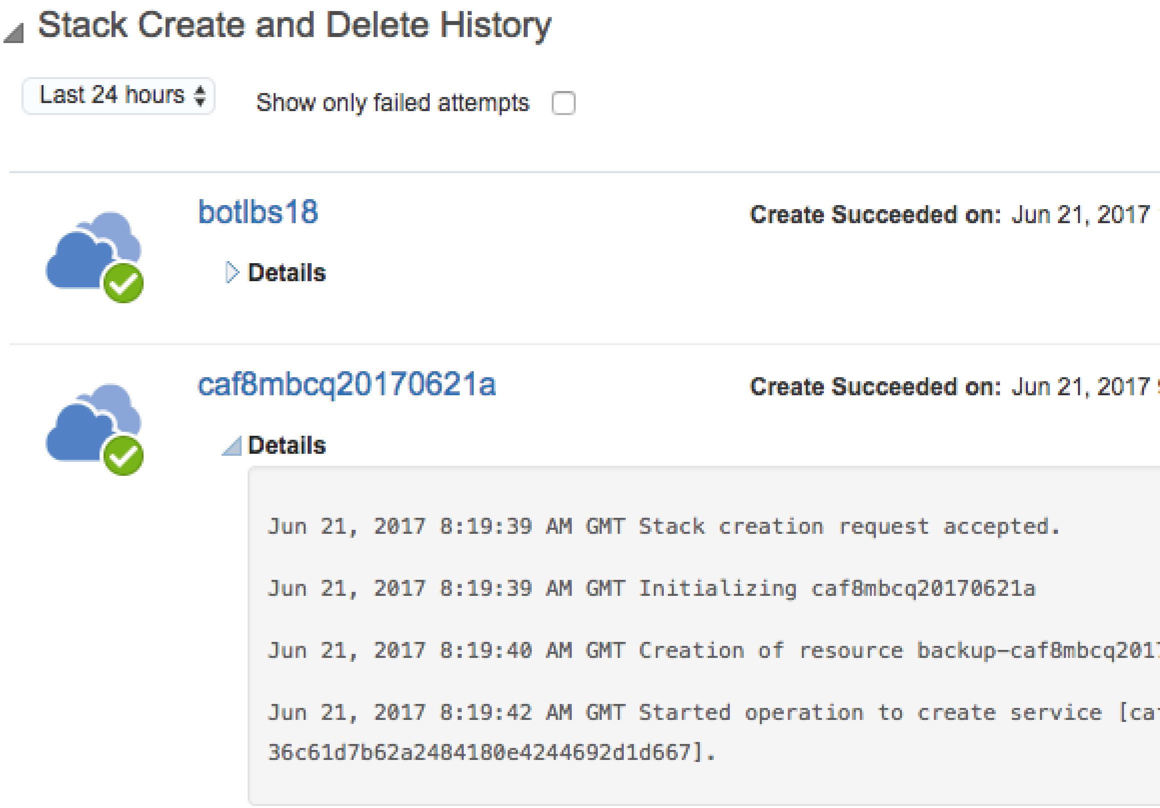
The service creation operation will clean up the failed service and its resources and retry service creation. It will do this twice, and if the failure has been caused by a temporary condition and if you wait you may see service creation succeed.

Service Creation Failure Analysis with the Stack Dashboard

On the Oracle Cloud Stack dashboard, open the **Stack Create and Delete History** pane. Click **Details** to see if the list of status messages give you an idea of why provisioning failed.

Possible reasons are:

- Maybe you have insufficient resources
- Perhaps you entered invalid data when you provisioned OMCe



botlbs18 **Create Succeeded on:** Jun 21, 2017

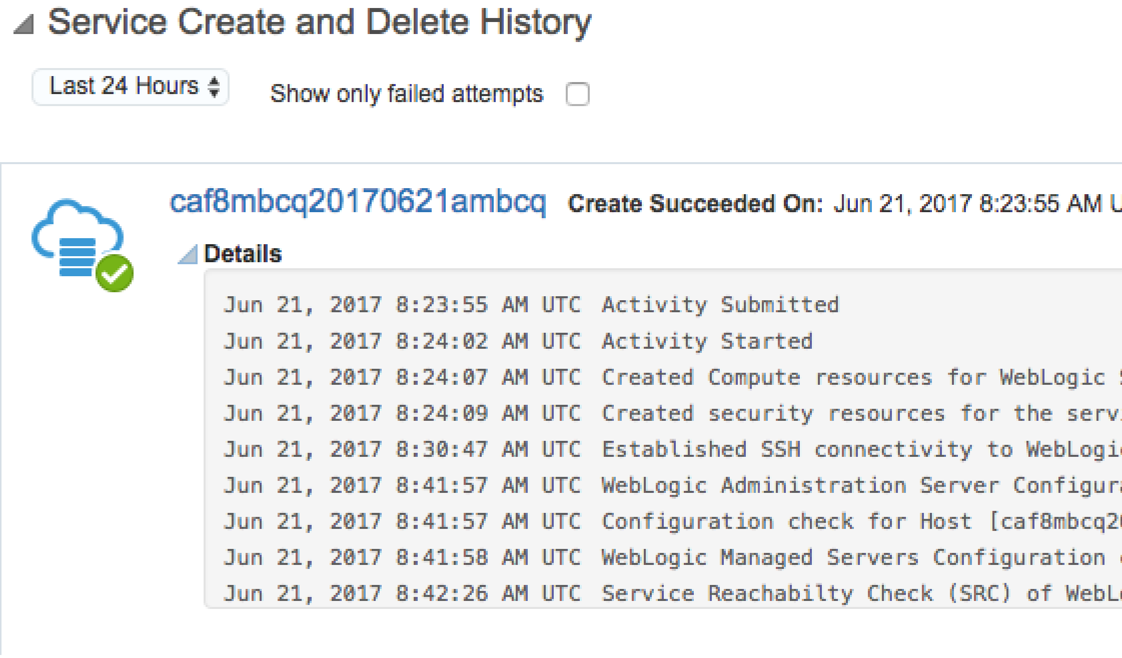
caf8mbcq20170621a **Create Succeeded on:** Jun 21, 2017

Details

```
Jun 21, 2017 8:19:39 AM GMT Stack creation request accepted.

Jun 21, 2017 8:19:39 AM GMT Initializing caf8mbcq20170621a

Jun 21, 2017 8:19:40 AM GMT Creation of resource backup-caf8mbcq201

Jun 21, 2017 8:19:42 AM GMT Started operation to create service [ca
36c61d7b62a2484180e4244692d1d667].
```

Service Creation Failure Analysis with the Service Dashboard

For an individual Service, such as Mobile Core, Bots or CxA, you can do the same at the Service level by navigating to the individual Service dashboard and opening the **Service Create and Delete** History pane near the bottom of the Stack Overview. Click **Details** to see why provisioning failed.



Service Create and Delete History
Last 24 Hours
Show only failed attempts □

caf8mbcq20170621ambcq **Create Succeeded On:** Jun 21, 2017 8:23:55 AM U

Details

```
Jun 21, 2017 8:23:55 AM UTC  Activity Submitted
Jun 21, 2017 8:24:02 AM UTC  Activity Started
Jun 21, 2017 8:24:07 AM UTC  Created Compute resources for WebLogic
Jun 21, 2017 8:24:09 AM UTC  Created security resources for the serv
Jun 21, 2017 8:30:47 AM UTC  Established SSH connectivity to WebLogi
Jun 21, 2017 8:41:57 AM UTC  WebLogic Administration Server Configur
Jun 21, 2017 8:41:57 AM UTC  Configuration check for Host [caf8mbcq2
Jun 21, 2017 8:41:58 AM UTC  WebLogic Managed Servers Configuration
Jun 21, 2017 8:42:26 AM UTC  Service Reachabilty Check (SRC) of WebL
```

Analytics Not Working

If there has been an issue during provisioning with passing the private key into the OMCe stack then Enrich, required by Analytics, may not be installed and so the Enrich Spark job will not run. You'll be able to see this because of the following effects:

- Analytics appears to function, but there are no events displayed.
- When you check the Big Data Service Console and you can see that the Enrich Spark job is not running.
- The `/tmp/id_rsa` file doesn't exist.

The solution is to perform the following tasks to input the private key, and to install Enrich and submit the Enrich Spark job.

Input the Private Key

- Copy the private key to the Analytics Pod Service VM:

  ```
  ssh -i privatekey opc@{PUBLIC_IP_CXAPOD_VM}
  ```

  Create the file `/tmp/id_rsa` and paste in private key content into it. Then execute the following:

  ```
  chmod 600 id_rsa
  chown oracle:oracle id_rsa
  ```

Install Enrich and Submit the Enrich Spark Job

1. Find the Analytics Pod Service VM's IP address.
   1. From OMCe Stack Overview, click the resource with the name `<stack>CXAPOD`. See View Your OMCe Stack.
   2. In the Service Overview for the Analytics Pod Service POD Details, expand Resources and note the Public IP, for example `192.0.2.1`.
2. Connect to the Core VM using ssh and change to the `oracle` user. You can log in as the default user, `opc`. The `opc` user has `sudo` privileges.
   1. Use:

      ```
      $ ssh -i <private_key_file> opc@<CXAPOD IP address>
      ```

      For example, `ssh -i my_private_rsa_keyfile opc@192.0.2.1`
   2. Switch to the `oracle` user using:

      ```
      $ sudo su - oracle
      ```
3. Change directories by executing:

   ```
   cd ~/../../cxpd/Manager/service_scripts/provisioning_check/
   ```

   Look for a file in this directory called `data_bag.json`. If it is present, continue with step 4.

   If the file doesn't exist in this directory, contact Oracle Support Services.
4. Install Enrich and submit the Enrich Spark job by executing:

   ```
   python check_provisioning_status_rex.py
   ```

Troubleshoot Service Creation Failure

If service creation fails during provisioning there are a couple of methods you can use to try to diagnose the cause of the failure.

Service Creation Failure Analysis with Mobile Doctor Provisioning Log Zip

You can use the Mobile Suite Doctor tool to download the provisioning dump. See Using the OMCe Mobile Suite Doctor.

Alternatively, you can use the Mobile Core Doctor to produce diagnostic log files which you can use to try to diagnose the root cause of the failure. See Using the OMCe Mobile Core Doctor. If you cannot diagnose the problem, you can contact Oracle Support and attach the zip file to a service request.

For Mobile Core Service (MobileCorePOD), the Mobile Core Doctor diagnostic dump is automatically uploaded to the storage container specified when OMCe is provisioned. The file name has the form `provisioning_logs/mbcp_postCreate_<timestamp>.zip`.

Download the zip file using cURL or ftmcli (Oracle Storage Cloud Service - File Transfer Manager Client). You can expand the zip file and see if you can diagnose the cause of the failure.

For information about downloading and installing cURL, see Update Settings for the Storage Container.

Operation	Command Format	Sample Output
List Mobile Doctor Dumps	`curl -X GET https://storage.oraclecorp.com/v1/<storage domain>/<container>?prefix=provisioning_logs -u <user>:<password>`	`provisioning_logs/mobdr_collection_20170621T065111.zip`
Download Mobile Doctor Dump	`curl -X GET https://storage.oraclecorp.com/v1/<storage domain>/<container>?prefix=provisioning_logs/<mobdr zip filename> -u <user>:<password> -o <mbcp zip filename>`	`provisioning_logs/mbcp_postCreate_20170621T065111.zip`

Service Creation Failure Analysis directly on the VM

You can SSH to the Mobile Core VM, Service Creation logs and other files helpful in diagnosing provisioning issues. They are available in the following locations:

- /u01/app/oracle/tools/paas/state/logs/rcu.results.txt
- /u01/app/oracle/tools/paas/state/logs/jaas.log
- /u01/app/oracle/tools/paas/state/logs/*_provisioning*
- /u01/app/oracle/tools/paas/state/logs/OraJCSmon

- /u01/app/oracle/tools/paas/state/work/jcs/OraJCSmon/records/JCSmon_*_domain_*
- /tmp/pod.postCreate.json
- /tmp/OraInstall*
- /tmp/wlstOfflineLogs_oracle
- /tmp/RCU*/logs/logger*.properties
- /tmp/RCU*/logs/rcu.log
- /tmp/mobile/patches/*/README.txt
- /tmp/wlstOfflineLogs_oracle/*.log

Troubleshoot Runtime Mobile Suite Issues

If you have a runtime failure, you can use the Mobile Suite Doctor tool to try to analyze the source of the problems. See Using the OMCe Mobile Suite Doctor.

- Query and download log files
- Do a full diagnostic dump at Suite level or Service level

Install the Mobile Suite Doctor, then follow these instructions to access the logs using the Oracle Storage Cloud Service (OSS) REST API. Log files are automatically uploaded to the OSS and stored as a series of zip files.

- You can query the set of log files and download them using the OSS REST API.
- You can trigger the upload of the latest logs to OSS using the PSM REST API

Troubleshoot Runtime Mobile Core Issues

If there are runtime failures in the Mobile Core, you can use the Mobile Core Doctor to try to diagnose the cause of the failure.

Runtime Failure Analysis using Mobile Core Doctor directly on the VM

You can use the Mobile Core Doctor to produce diagnostic log files which you can use to try to diagnose the root cause of the failure. See Using the OMCe Mobile Core Doctor. If you cannot diagnose the problem, you can contact Oracle Support and attach the zip file to a service request.

For Mobile Core Service (MobileCorePOD), the Mobile Core Doctor diagnostic dump is automatically uploaded to the storage container specified when OMCe was provisioned. It has the name `provisioning_logs/mobdr_collection_`*`timestamp`*`.zip`.

Using the OMCe Mobile Suite Doctor

If you have to contact Oracle Support about a problem with OMCe they may ask you to use the Mobile Suite Doctor to collect useful diagnostic and configuration information from Mobile Suite logs and attach it to a service request.

Download and install the Mobile Suite Doctor

The Mobile Suite Doctor is part of the Admin tools zip which you can get from the OMCe Downloads page on OTN at http://www.oracle.com/technetwork/topics/cloud/downloads/mobile-suite-3636471.html. Save the zip to a local location and unzip it.

You have to have Python 3 installed on the same machine where you install and run Mobile Suite Doctor. You can get it from https://www.python.org/downloads/.

> **Note:**
>
> The Mobile Suite Doctor uses the executable `python`. If the version of Python 3 you have has the `python3` executable you should either modify the `omcedr.sh/omcedr.cmd` scripts to use `python3`, or symlink `python` to the `python3` executable.
>
> On OSX, if you get an error `SSL: CERTIFICATE_VERIFY_FAILED` when you run the tool it is probably because Python 3.6 on OSX has no certificates and can't validate any SSL connections. You can install the `certifi` package of certificates. See the ReadMe at `/Applications/Python\ 3.6/ReadMe.rtf` .

Using a Configuration File for Properties

The Mobile Suite Doctor uses information about the service and the Storage Service that it will run against. You can use a configuration file to pass these properties and their values to the tool. If you choose not to use a configuration file, you will be prompted for the values as the tool runs. You can choose to have some of the information in the configuration file and enter other values when prompted. This allows you to keep sensitive information such as passwords secure.

Create a JSON file to contain some or all of the following properties and values. The default name of the file is `omcedr.json` in the same directory as `omcedr`. You can choose to use another filename name or location using the command line option `–omcedr_config_file`.

Name	Description	Example
`psmUser`	The username for My Services and the Service Console	`mary.jones@greencorp.com`
`psmPassword`	The password for My Services and the Service Console	*`password`*
`psmUri`	The URI for My Services. Use the PSM URI for your datacenter region: • `psm.us.oraclecloud.com` • `psm.europe.oraclecloud.com` • `psm.aucom.oraclecloud.com` • `psm.lad.oraclecloud.com`	`https://psm.us.oraclecloud.com`

Name	Description	Example
`psmIdentityDomain`	The identity domain	`idcs-e8917a96fe5464d9a893eb90be983f18`
`storageUser`	Name of the Storage user	`dae78aaf9b48be68a1250b249f5af383`
`storagePassword`	Password for the Storage user	*`password`*
`storageUri`	Mobile Suite Container Storage URI	`https://storage.oraclecorp.com/v1/Storage-dae78aaf9b48be68a1250b249f5af383/OmceContainer`

omcedr Log File

The Mobile Suite Doctor generates its own log file, `omcedr.log`, that contains information and error messages generated by the tool. Mobile Suite Doctor can also generate detailed logs using the command line option `-ll` to use `DEBUG` level.

omcedr

Use to see the full syntax and help for running each of the command options.

Syntax

```
omcedr.sh
```

Parameters

All parameters are required unless otherwise noted.

Parameter	Description
`--help`	Displays the full help for the Mobile Suite Doctor. By default only the syntax is displayed

omcedr list_logs

Use to get a list of available logs either for the entire Mobile Suite or a single Service.

Syntax

```
omcedr.sh list_logs -s|--stack stackName [-n|--service_name serviceName] [-ft|--from_time fromTime] [-tt|--to_time toTime] [-f|--omcedr_config_file configFile] [-ll|--log_level logLevel]
```

Parameters

All parameters are required unless otherwise noted.

Parameter	Description
`-s|–stack` *`stackName`*	Mobile Suite stack name
`-n|–service_name` *`serviceName`*	(Optional) Mobile Suite service name

Parameter	Description
-ft|–from_time *fromTime*	(Optional) Start time period to query the list of logs, based on the log file last update time. Default 24 hours ago. Date format: YYYY-MM-DD HH:MM (e.g. 2017-07-26 20:04)
-tt|–to_time *toTime*	(Optional) End time period to query the list of logs, based on the log file last update time. Default is the current date and time. Date format: YYYY-MM-DD HH:MM (e.g. 2017-07-27 20:04)
-f|–omcedr_config_file *configFile*	(Optional) Configuration file
-ll|-log_level *logLevel*	(Optional) Log level to set for the tool's internal logging (WARNING, INFO, DEBUG). Default is INFO.

omcedr push_logs

Use to force the ACCS logs to be pushed to OSS for the entire Mobile Suite or a single Service.

Syntax

```
omcedr.sh push_logs -s|--stack stackName [-n|--service_name serviceName] [-f|--
omcedr_config_file configFile] [-ll|--log_level logLevel]
```

Parameters

All parameters are required unless otherwise noted.

Parameter	Description
-f|–omcedr_config_file *configFile*	(Optional) Configuration file
-ll|-log_level *logLevel*	(Optional) Log level to set for the tool's internal logging (WARNING, INFO, DEBUG). Default is INFO.
-n|–service_name *serviceName*	(Optional) Mobile Suite service name
–s|–stack stackName	Mobile Suite stack name

omcedr download_logs

Use to download logs either for the entire Mobile Suite or a single Service. It is also possible to download a named single file. The downloaded logs will be written to a temporary directory.

Syntax

```
omcedr.sh download_logs -s|--stack stackName [-n|--service_name serviceName] [-
ft|--from_time fromTime] [-tt|--to_time toTime] [-f|--omcedr_config_file
configFile] [-l|--log logFilename] [-od|-output_directory] [-ll|--log_level
logLevel]
```

Parameters

All parameters are required unless otherwise noted.

Parameter	Description
-s|–stack *stackName*	Mobile Suite stack name
-n|–service_name *serviceName*	(Optional) Mobile Suite service name
-ft|–from_time *fromTime*	(Optional) Start time period to download logs, based on the log file last update time. Default 24 hours ago. Date format: YYYY-MM-DD HH:MM (e.g. 2017-07-26 20:04)
-tt|–to_time *toTime*	(Optional) End time period to download logs, based on the log file last update time. Default is the current date and time. Date format: YYYY-MM-DD HH:MM (e.g. 2017-07-27 20:04)
-l|–log *logFilename*	(Optional) Full path and name of log file to download
-f|–omcedr_config_file *configFile*	(Optional) Configuration file with the PSM and OSS settings
-ll|-log_level *logLevel*	(Optional) Log level to set for the tool's internal logging (WARNING, INFO, DEBUG). Default is INFO.
-od|-output_directory *outputDirectory*	(Optional) Directory to write the logs to. Default is the system's "temp" directory.

omcedr check_health

Use this command to perform a health check for a named service.

Syntax

```
omcedr.sh check_health -n|--service_name serviceName -t|--service_type serviceType
[-f|--omcedr_config_file configFile] [-ll|--log_level logLevel]
```

Parameters

All parameters are required unless otherwise noted.

Parameter	Description
-n|–service_name*serviceName*	Mobile Suite service name
-t| --service_type*serviceType*	Mobile Suite service type (e.g. `MobileCorePOD`)
-f|–omcedr_config_file*configFile*	(Optional) Configuration file with the PSM and OSS settings
-ll|-log_level logLevel	(Optional) Log level to set for the tool's internal logging (`WARNING`, `INFO`, `DEBUG`). Default is `INFO`.

omcedr dump

Use to perform a diagnostic dump for the entire Mobile Suite or a single Service. The output from running this command is a zip file containing the captured diagnostics information.

Syntax

```
omcedr.sh dump -s|--stack stackName [-n|--service_name serviceName] [-ft|--
from_time fromTime] [-tt|--to_time toTime] [-pd|--provisioning_dumps] [-f|--
omcedr_config_file configFile] [-od|--output_directory outputDirectory] [-ll|--
log_level logLevel]
```

Parameters

All parameters are required unless otherwise noted.

Parameter	Description
-s|–stack *stackName*	Mobile Suite stack name
-n|–service_name *serviceName*	(Optional) Mobile Suite service name
-ft|–from_time *fromTime*	(Optional) Start time period to download logs, based on the log file last update time. Default 24 hours ago. Date format: YYYY-MM-DD HH:MM (e.g. 2017-07-26 20:04)
-tt|–to_time *toTime*	(Optional) End time period to download logs, based on the log file last update time. Default is the current date and time. Date format: YYYY-MM-DD HH:MM (e.g. 2017-07-27 20:04)
-l|–log *logFilename*	(Optional) Full path and name of log file to download
-f|–omcedr_config_file *configFile*	(Optional) Configuration file with the PSM and OSS settings
-ll|-log_level *logLevel*	(Optional) Log level to set for the tool's internal logging (WARNING, INFO, DEBUG). Default is INFO.
-od|-output_directory *outputDirectory*	(Optional) Directory to write the logs to. Default is the system's "temp" directory.

omcedr stack_info

Use to get information from PSM about a Mobile Suite stack.

Syntax

```
omcedr.sh stack_info -s|--stack stackName [-f|--omcedr_config_file configFile] [-
ll|--log_level logLevel]
```

Parameters

All parameters are required unless otherwise noted.

Parameter	Description
-s|–stack *stackName*	Mobile Suite stack name
-f|–omcedr_config_file *configFile*	(Optional) Configuration file with the PSM and OSS settings
-ll|-log_level *logLevel*	(Optional) Log level to set for the tool's internal logging (WARNING, INFO, DEBUG). Default is INFO.

omcedr service_info

Use to get information from PSM about a specific Mobile Suite service.

Syntax

```
omcedr.sh service_info -n|--service_name serviceName -t|--service_type serviceType
[-f|--omcedr_config_file configFile] [-ll|--log_level logLevel]
```

Parameters

All parameters are required unless otherwise noted.

Parameter	Description
`-n|–service_name` *`serviceName`*	Mobile Suite service name
`-t| --service_type` *`serviceType`*	Mobile Suite service type (e.g. MobileCorePOD)
`-f|–omcedr_config_file` *`configFile`*	(Optional) Configuration file with the PSM and OSS settings
`-ll|-log_level` *`logLevel`*	(Optional) Log level to set for the tool's internal logging (WARNING, INFO, DEBUG). Default is INFO.

Using the OMCe Mobile Core Doctor

If you have to contact Oracle Support about a problem with OMCe they may ask you to use the Mobile Core Doctor to collect diagnostic information from the mobile core host VM and attach it to a service request.

Overview

You can run the Mobile Core Doctor on the mobile core host VM.

The Mobile Core Doctor also runs automatically at various points in the core service’s lifecycle. The output zip files are uploaded to the storage account and container associated with the service.

Lifecycle Point	Collections Performed	Cloud Storage Filename
Post-Create	local_provisioning, local_offline_runtime, wls_online_state, wls_online_health	`provisioning_logs/mbcp_postCreate_<timestamp>.zip`
Pre-Delete	local_provisioning, local_offline_runtime, wls_online_state, wls_online_health	`provisioning_logs/mbcp_preDelete_<timestamp>.zip`

The timestamp is of the form `<YEAR><MONTH><DAY>T<HOUR><MIN><SECONDS>`: (e.g. `20170911T105647`).

Run the Mobile Core Doctor and Copy Log Files

If you have been asked by support to attach log files from the Mobile Core Doctor to a service request, follow the steps below.

1. Find the Core VM’s IP address.
   a. From OMCe Stack Overview, click the resource with the name `<stack>CORE`. See View Your OMCe Stack.

   b. In the Service Overview for the Mobile Core POD Details, expand Resources and note the Public IP, for example `192.0.2.1`.

2. Connect to the Core VM using ssh and change to the `oracle` user. You can log in as the default user, `opc`. The `opc` user has `sudo` privileges.

   a. To make the private key not user-readable, use:

   ```
   chmod 600 <private_key_file>
   ```

   b. Log in using:

   ```
   $ ssh -i <private_key_file> opc@<Core IP address>
   ```

   For example, `ssh -i my_private_rsa_keyfile opc@192.0.2.1`

   c. Switch to the `oracle` user using:

   ```
   $ sudo su - oracle
   ```

3. Run the Mobile Core Doctor to generate the log file zip.

   a. Use `cd /u01/app/oracle/suite/mcs/mobile_doctor`.

   b. Run `mobile_doctor.sh collect`. You might be prompted to provide the username, password and URI for My Services. For details on commands and parameters, see Mobile Core Doctor Tool Options.

   A zip file is generated under `/tmp`. The name of the file is `<prefix>_<timestamp>.zip`, for example, `/tmp/mobdr_collection_20170707T100806.zip`

4. Copy the zip file from the Core VM using scp. You can use a tool such as pscp (which comes with PUTTY) which allows you to copy files from and to remote machines. Use:

   ```
   $ scp -i <private_key_file> opc@<Core IP address>:/tmp/file.zip file.zip
   ```

   where *`file.zip`* is the zip file generated in the previous step. For example:

   ```
   $ scp -i ~/.ssh/my_private_rsa_keyfile opc@192.0.2.1:/tmp/
   mobdr_collection_20170707T100806.zip mobdr_collection_20170707T100806.zip
   ```

   > **Note:**
   >
   > You might need to change the commands above depending on the scp tool you use. For example, if you are using pscp you would start the command with `$ pscp`.

Mobile Core Doctor Tool Options

The Mobile Core Doctor Tool is located on the Mobile Core VM (`/u01/app/oracle/suite/mcs/mobile_doctor/mobile_doctor.sh`).

mobile_doctor.sh

When run without any options the tool will print information about the available options and their sub-options.

Syntax

```
mobile_doctor.sh
```

Parameters and Options

None

mobile_doctor.sh list_collections

Display the list of known collections and the summary description of each of them.

Syntax

```
mobile_doctor.sh list_collections
```

Parameters and Options

None

Details of the Collections

- **local_offline_runtime**

  This collection focuses on data that the WebLogic domain reads or writes to disk. It includes certain configuration files, log files, incidents and flight recordings. It tends to be the flight recordings (binary, don't compress well) that take up most space in the mobile doctor output, and the script might not collect them if they are too big.
- **local_provisioning**

  This collection focuses on the log files that result from provisioning mobile core components on the core host.
- **wls_online_state**

  This collection uses WLST to fetch various pieces of information from the WLS MBeans that represent the domain, servers, applications and data-sources hosted by that WLS instance.
- **wls_online_health**

  This collection uses WLST to inspect the runtime status of servers, applications and data-sources required to run mobile core.

mobile_doctor.sh collect

Collects the information and collates it in a zip file in the temporary directory on the host (typically `/tmp`). The name of the resulting zip file is of the form `<prefix>_<timestamp>.zip`, for example, `/tmp/mobdr_collection_20170714T095656.zip`.

You might be prompted to provide the username, password and URI for My Services.

Syntax

```
mobile_doctor.sh collect [maxsize=<size>] [fileprefix=<prefix>]
[collection_names...]
```

Parameters and Options

Parameter/Option	Mandatory/Optional	Description
maxsize	Optional Default: 25M	The approximate maximum size of the zip file to be created. As the mobile doctor collects information it will constantly check the size of the zip file and predict if the next piece of content will push the zip file over the prescribed limit. If it will then that piece of content will not be recorded, but the tool will go on to try and collect other pieces of data.
fileprefix	Optional Default: mobdr_collection	Specifies the prefix of the resulting zip file
collection_names	Optional Default: local_offline_runtime	The names of one or more of the predefined collections.

Getting Log Messages

If you have to contact Oracle Support Services, they may ask you to download log messages to send to them. For example, Oracle Support may ask you to get the core and custom code logs. This section describes how to do this.

Which Tool to Use

There are two diagnostic tools

- Mobile Suite Doctor, which you use to get the MobileCCC, Bots, and CxA logs. See Using the OMCe Mobile Suite Doctor.
- Mobile Core Doctor, which you use to get the mobile logs. See Using the OMCe Mobile Core Doctor.

The procedures you should expect to run are these:

1. Install and run the Mobile Suite Doctor, omcedr, to collect MobileCCC, Bots, and CxA logs, described inUsing the OMCe Mobile Suite Doctor:
   a. Install the Mobile Suite Doctor.
   b. Define a configuration file.
   c. Run the `download_logs` command. Ensure that `from_time` and `to_time` are before and after the period of the issue that you are reporting so that any events will be captured in the messages.
2. Follow the procedure in Using the OMCe Mobile Core Doctor to use the Mobile Core Doctor to collect mobile logs.
   a. Use ssh to connect to the Mobile Core Doctor.
   b. Run the tool with the single `collect` argument to generate a zip file.
   c. Copy the zip file from the Core VM using scp.
3.

9

Managing OMCe with the PaaS CLI

As an alternative to using the PaaS Service Manager UI and Oracle Cloud Stack Manager UI to manage the lifecyle of your OMCe service instance and stack, respectively, you can perform the same lifecycle operations with the PaaS Service Manager command line interface (CLI). The CLI is a thin wrapper over PaaS REST APIs that you can use to run command to create, start, stop, apply patches, backup, restore, and delete a service.

Before you can use the PaaS CLI, you must have the cURL command-line tool and Python 3.3 or later. To get the PaaS CLI, you'll have to perform the following actions:

Action	How Do I Do It?
Download the CLI	To download from the Oracle Cloud User Interface, see Downloading the CLI from the Oracle Cloud User Interface in the *PaaS Service Manager Command Line Interface Reference*. To download using a REST API, see Downloading the CLI by Using the REST API in the *PaaS Service Manager Command Line Interface Reference*.
Install the CLI	See Installing the Command Line Interface in the *PaaS Service Manager Command Line Interface Reference*.
Configure the CLI	See Configuring the Command Line Interface in the *PaaS Service Manager Command Line Interface Reference*.

Some common Oracle Cloud Stack Manager commands are listed here.

Stack Command	Description
`psm stack create`	Create a stack.
`psm stack delete`	Delete a stack.

`psm stack start`
`psm stack stop`
	Start or stop all resources in the specified stack.

`psm stack describe`	List detailed information about a stack.
`psm stack list`	List all stack instances in the identity domain.
`psm stack resume`	Resume creating a stack. Use this command when creation of a stack was stopped when an error occurred and the stack creation wasn't rolled back.
`psm stack operation-status`	List detailed information about a stack operation, particularly a stack creation operation.

Some common service commands are listed here.

Service Instance Command	Description
`psm` *`omce`* `services`	List all active service instances within your identity domain. Set the output level to `verbose` to see all the details about each instance.
`psm` *`omce`* `service`	Display the details for a specified service.

`psm` *`omce`* `start`
`psm` *`omce`* `stop`
	Start or stop the OMCe service instance.

`psm` *`omce`* `restore`	Restore a OMCe service instance from the specified backup.
`psm` *`omce`* `backup`	Initiate an on-demand backup for the OMCe service instance.
`psm` *`omce`* `patch`	Apply a patch to the specified OMCe service instance.
`psm` *`omce`* `available-patches`	List all available patches for an OMCe service instance.
`psm` *`omce`* `applied-patches`	List all applied patches to an OMCe service instance.
`psm` *`omce`* `rollback`	Rollback a patch for the specified OMCe service instance.
`psm` *`omce`* `restart`	Restart an OMCe administration server.

Replace *omce* with the appropriate service type from this list:

- `BOTSCFG`
- `BOTSINT`
- `BOTSCON`
- `BOTSPIP`
- `BOTSMGM`
- `OEHCS`
- `CXACFG`
- `CXAPOD`
- `CXAANA`
- `CXACOL`
- `MobileCorePOD`
- `MobileCCC`

See Oracle Cloud PaaS Service Manage Command Line Interface Reference for a complete list of OMCe commands, their syntax, and parameters.

Monitor Your OMCe Stack with the CLI

You can monitor the OMCe stack with the Oracle Cloud Stack Manager console or you can monitor it from the command line:

1. Run the `stack list` command.

   Use the output to identify the names of your cloud stacks. For example:

```
psm stack list

{
  "identityDomain":"MyIdentityDomain",
  "stacks":[
     {
          "serviceName":"MyStack1",
          "description":"",
          "serviceURI":"http://example.org:7200//paas/api/v1.1/instancemgmt/
MyIdentityDomain/services/stack/instances/MyStack1",
          "state":"READY",
          "stateDetail":"Ready",
          "identityDomain":"MyIdentityDomain",
          "createdBy":"joe@example.com",
          "creationJobId":"80161",
          "creationTime":"2016-07-26T15:48:18.366+0000",
          "lastModifiedTime":"2016-07-26T15:48:18.365+0000",
          "template":{
          "templateName":"MyTemplate",
          "templateVersion":"1.0.0",
          "templateURI":"http://example.org:7200//paas/api/v1.1/
instancemgmt/MyIdentityDomain/templates/cst/instances/MyTemplate/export?
version=1.0.0"
          }
       },
       {
          "serviceName":"MyStack2",
          "description":"",
          "serviceURI":"http://example.org:7200//paas/api/v1.1/instancemgmt/
MyIdentityDomain/services/stack/instances/MyStack1",
          "state":"READY",
          "stateDetail":"Ready",
          "identityDomain":"MyIdentityDomain",
          "createdBy":"joe@example.com",
          "creationJobId":"80161",
          "creationTime":"2016-07-27T15:48:18.366+0000",
          "lastModifiedTime":"2016-07-27T15:48:18.365+0000",
          "template":{
          "templateName":"MyTemplate",
          "templateVersion":"1.0.0",
          "templateURI":"http://example.org:7200//paas/api/v1.1/
instancemgmt/MyIdentityDomain/templates/cst/instances/MyTemplate/export?
version=1.0.0"
            }
        }
    ]
}
```

2. Use the `stack describe` command:

   To get details about a specific stack `psm stack describe -n MyStack1`

   To show all details of the stack, append `–expand all`

Start or Stop a Stack with the CLI

Besides using the Oracle Cloud Stack Manager console, you can also start or stop all the resources in a stack from the PSM Command Line Interface (CLI).

1. Run the `stack start` or `stack stop` command and provide the name of the stack:

   ```
   psm stack start –n name
   ```

   ```
   psm stack stop –n name
   ```

2. Periodically check the progress of the start or stop operation:

   ```
   psm stack operation-status –j jobID
   ```

   When you start or stop a job, the output includes the job ID number.

Delete a Stack with the CLI

You can delete a stack from the command line:

1. Run the `stack delete` command and include the name of the stack:

   ```
   psm stack delete -n name
   ```

2. Check the progress of the delete operation by running the `stack operation` command and providing the job ID:

   ```
   psm stack operation -j jobID
   ```

Monitor a Service Instance with the PaaS CLI

Display details of a single OMCe service instance within your identity domain with the `psm omce service` command.

This command takes the following parameters:

Parameter	Description
`-s|--service-name` *`serviceName`*	Specifies the instance.
`-of|--output-format json|html|short`	(Optional) Specifies the output format. You can specify which format to use as the default with the `psm setup` command. See Configuring the Command Line Interface in the *Oracle Cloud PaaS Service Manage Command Line Interface Reference*.

Display a list of all active OMCe service instances within your identity domain with the `psm omce services` command.

This command takes the following parameters:

Parameter	Description
`-o|--output–level verbose`	(Optional) Displays the full set of details in the output for each service instance.

Parameter	Description
`-of|--output-format json|html|short`	(Optional) Specifies the output format. You can specify which format to use as the default with the `psm setup` command. See Configuring the Command Line Interface in the *Oracle Cloud PaaS Service Manage Command Line Interface Reference*

Precheck Before Patching with the PaaS CLI

Prior to patching an OMCe service instance, run the `psm omce precheck-patch` command to identify potential issues that could prevent the patch operation from completing. This command returns reports on

- disk space shortage
- database connectivity failure
- server access failure
- storage access failure

This command doesn’t check if other administrative operations, such as backup or restore, are in progress even though the operations would prevent a patch operation from completing.

This command takes the following parameters:

Parameter	Description
`-s|--service-name` *`serviceName`*	Specifies the instance.
`-p--patch–id` *`patchId`*	Specifies the patch Id for the patch to precheck.
`-of|--output-format json|html|short`	(Optional) Specifies the output format. You can specify which format to use as the default with the psm setup command. See Configuring the Command Line Interface in the *Oracle Cloud PaaS Service Manage Command Line Interface Reference*.

Patch an OMCe Service with the PaaS CLI

Patch an OMCe service instance with the `psm omce patch` command.

This command takes the following parameters:

Parameter	Description
`-s|--service-name` *`serviceName`*	Specifies the instance.
`-p--patch–id` *`patchId`*	Specifies the patch. Use the `available-patch` command to retrieve the patch Id.
`-n|--additional-note` *`free form note text`*	(Optional) Provides additional information about the patch in free-form text.

Parameter	Description
`-of|--output-format json|html|short`	(Optional) Specifies the output format. You can specify which format to use as the default with the psm setup command. See Configuring the Command Line Interface in the *Oracle Cloud PaaS Service Manage Command Line Interface Reference*.

Rollback a Patch with the PaaS CLI

If you need to roll back a patch, use the `psm omce rollback` command.

This command takes the following parameters.

Parameter	Description
`-s|--service-name` *`serviceName`*	Specifies the instance.
`-r--rollback-id` *`nn`*	Identifies the patch operation to roll back. Use the `applied–patches` command to retrieve the rollback Id.
`-n|--additional-note` *`free form note text`*	Provides additional information about the patch rollback in free-form text.
`-of|--output-format json|html|short`	(Optional) Specifies the output format. You can specify which format to use as the default with the psm setup command. See Configuring the Command Line Interface in the *Oracle Cloud PaaS Service Manage Command Line Interface Reference*.