Oracle® Database
Administering Oracle Blockchain Platform
This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.
Contents

Preface

Audience v
Documentation Accessibility v
Related Documents v
Conventions v

1 A Service Administrator's Roadmap to Oracle Blockchain Platform

Oracle Blockchain Platform Enterprise Edition Overview 1-1
Security, Authentication, and Authorization 1-2
Workflow for Administering Oracle Blockchain Platform 1-6

2 Design Your Oracle Blockchain Platform Configuration

Prerequisites 2-1
Supported Topologies 2-2

3 Install Your Oracle Blockchain Platform Instance

Deploy Your Virtual Machine 3-1
Log on to Oracle Blockchain Platform for the First Time 3-2

4 User Management

Configure an LDAP Server 4-1
  Configure the Built-In LDAP Server 4-1
  Configure an External LDAP Server 4-3
Add Users to Your LDAP Server 4-4
User Roles 4-5

5 Provision an Instance

Before You Create an Oracle Blockchain Platform Instance 5-1
Provision an Instance using the Blockchain Platform Manager 5-2
Provision an Instance Using REST APIs 5-3
Postrequisites When Using an External Load Balancer 5-5

6 Manage Oracle Blockchain Platform

View Instance Details 6-1
Start or Stop an Instance 6-1
Delete an Instance 6-1
Scale an Instance In or Out 6-2
Patch an Instance 6-2

7 Monitor and Troubleshoot Your Instance

Logging 7-1

A Accessibility Features and Tips for Oracle Blockchain Platform
Preface

*Administering Oracle Blockchain Platform* explains how to provision and maintain Oracle Blockchain Platform instances.

**Topics:**
- Audience
- Documentation Accessibility
- Related Documents
- Conventions

**Audience**

This guide is intended for service administrators responsible for provisioning and maintaining Oracle Blockchain Platform.

**Documentation Accessibility**


**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](http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info) or visit [http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs](http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs) if you are hearing impaired.

**Related Documents**

For more information, see these Oracle resources:
- *Using Oracle Blockchain Platform*

**Conventions**

The following text conventions are used in this document:
<table>
<thead>
<tr>
<th>Convention</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>boldface</strong></td>
<td>Boldface type indicates graphical user interface elements associated</td>
</tr>
<tr>
<td></td>
<td>with an action, or terms defined in text or the glossary.</td>
</tr>
<tr>
<td><em>italic</em></td>
<td>Italic type indicates book titles, emphasis, or placeholder variables</td>
</tr>
<tr>
<td></td>
<td>for which you supply particular values.</td>
</tr>
<tr>
<td><strong>monospace</strong></td>
<td>Monospace type indicates commands within a paragraph, URLs, code</td>
</tr>
<tr>
<td></td>
<td>in examples, text that appears on the screen, or text that you enter.</td>
</tr>
</tbody>
</table>


A Service Administrator's Roadmap to Oracle Blockchain Platform

Topics

- Oracle Blockchain Platform Enterprise Edition Overview
- Security, Authentication, and Authorization

Oracle Blockchain Platform Enterprise Edition Overview

Oracle Blockchain Platform gives you a pre-assembled platform for building and running smart contracts and maintaining a tamper-proof distributed ledger.

Oracle Blockchain Platform is a network consisting of validating nodes (peers) that update the ledger and respond to queries by executing smart contract code—the business logic that runs on the blockchain. External applications invoke transactions or run queries through client SDKs or REST API calls, which prompts selected peers to run the smart contracts. Multiple peers endorse (digitally sign) the results, which are then verified and sent to the ordering service. After consensus is reached on the transaction order, transaction results are grouped into cryptographically secured, tamper-proof data blocks and sent to peer nodes to be validated and appended to the ledger. Platform administrators can use the Blockchain Platform Manager to create and manage platform instances, while network administrators can use the Oracle Blockchain Platform console to configure the blockchain and monitor its operation.

Oracle Blockchain Platform Enterprise Edition provides an independently-installable version of Oracle Blockchain Platform built on Docker containers and delivered as a pre-built VM image for multiple virtualization options. The VM is delivered in an Open Virtualization Appliance (ova) format and can be imported and started using VMWare ESXi, Oracle VirtualBox, and Oracle Linux Virtualization Manager. Once the VM is running, the Blockchain Platform Manager is used for configuration, provisioning, and patching multiple Blockchain Platform instances, which can be deployed over multiple VMs to distribute the Docker containers running Oracle Blockchain Platform nodes. Similarly to the cloud PaaS, this edition is fully pre-assembled and can create new complete blockchain instances in minutes.

In addition to flexible virtualization options, the enterprise edition enables dynamic scalability to handle the evolving workloads by increasing the resources in the current VMs or scaling out to more VMs to run the additional nodes (e.g., peers, orderers.) Additional VMs and nodes can be deployed in other datacenters across a WAN for disaster recovery (DR.) Unlike typical applications, Oracle Blockchain Platform’s distributed ledger and the distributed metadata database handle data replication out-of-the-box.

Feature parity with the cloud version ensures that customers can deploy chaincode and use the same chaincode APIs and extensive REST APIs across both versions. Oracle innovations in using Berkeley DB for world state with SQL-based queries, built-in transaction synchronization to off-chain rich history database, intuitive and
comprehensive console with powerful operations and monitoring tools, and all the other unique enterprise-grade features are shared across the cloud and on-premise versions.

Security, Authentication, and Authorization

Introduction to Oracle Blockchain Platform Enterprise Edition Security

Oracle Blockchain Platform Enterprise Edition deals with security on several levels. At the top level is the security related to the Oracle Blockchain Platform virtual machines (VMs). Next is the security associated with the control plane that is used to manage the life cycle on Oracle Blockchain Platform instances. Control plane (the Blockchain Platform Manager) users are able to create, scale out, scale in, patch, and other life cycle operations. For each instance there are users authorized for managing, monitoring, and administering an instance. Finally there are users of the instance that access an instance either via the Fabric SDK or the Oracle Blockchain Platform REST Proxy.

All user information including their roles and passwords are stored in LDAP. A default LDAP server is provided as part of the VM and is only intended for development purposes. It is expected that you will connect to your corporate LDAP provider in production.

Managing Security

Creating Oracle Blockchain Platform VMs

Oracle Blockchain Platform uses a cluster of VMs based upon the VM image provided. The first step that needs to be taken in configuring Oracle Blockchain Platform VMs is to import them into whatever hypervisor is being used. Supported hypervisors include:

- VirtualBox – Primarily intended for development and testing
- VMware vSphere

Securing Data at Rest

At the time of importing the OVA file or sometime later users may want to enable disk encryption in their hypervisor to protect data at rest. This may also require that the VM be encrypted, which would also be handled by hypervisor settings.

Control Plane VM and Data Plane Clusters

A particular VM instance should be reserved for the Oracle Blockchain Platform control plane. This VM will run the provisioning server that is used to control the life cycle operations of Oracle Blockchain Platform instances within a specific Oracle Blockchain Platform platform. It is possible to deploy multiple Oracle Blockchain Platform platforms, each with its own control plane VM and clusters of data plane VMs. Each cluster of data plane VMs only supports a single Oracle Blockchain Platform instance.
Data plane clusters are automatically added to the Docker swarm used to allow containers to communicate with each other.

**Securing the VM Network**

You’ll initially log in with user ID `oracle`, and change your password as described in *Add Users to Your LDAP Server*. After logging in, configure the firewall (`firewalld`) on the VM to protect ports that shouldn’t be exposed outside the cluster. Whitelists should be used to only allow access to and from the other VMs in the cluster, and to grant access from any external load balancers that may be used.

The only network ports data plane VMs need to have accessible outside their cluster is for access to:

- Data plane console server
- Fabric-CA
- Ordering Service Nodes
- Peers

These ports should all be in the 10000-10200 range. No other ports should be accessible from outside the data plane cluster. These ports will automatically be exposed outside the VM at provisioning time.
Figure 1-2  Firewall, load balancer, and instance relationship

Normal Cluster Members

The VMs in a cluster need to be able to communicate with each other via Docker swarm. See the section below on which ports need to be accessible to other VMs in the cluster.

CRC Cluster Members

In production deployments, it is highly recommended that the chaincode runtime containers be placed in their own VM. Unless your chaincode needs to access external services, the VM should be configured to only be able to communicate with the other members of the cluster.

SSH Configuration

As delivered from Oracle, the VM appliance is configured with a default user of oracle and a default password of Welcome1. The password must be changed upon first login. In order to manage the VM, SSH is allowed with password-based authentication. This should be changed to public key based authentication for user oracle. No other users are required or should be configured.

VM Network Configuration

The Oracle Blockchain Platform VMs come configured and ready to run. The following ports are open on the VM. Most of these ports should not be exposed to the Internet or other unsecured hosts and access should be blocked by firewall rules on the VMs.

Internet Accessibility
Some ports may need to be accessible outside the corporate network. In particular, to have other Oracle Blockchain Platform or Fabric instances running outside the corporate network connect to an instance inside the corporate network, certain ports associated with the ordering service and peers will need to be accessible.

**Corporate Network Accessibility**

It is recommended that console UI ports associated with the provisioning console and instance console be restricted to at most access from inside the corporate network. Ideally they would be restricted to even a subset of that, only machines used for network management and operations.

**Port Accessibility Guide**

<table>
<thead>
<tr>
<th>Port</th>
<th>Use or purpose</th>
<th>Internet accessible</th>
<th>Corporate network accessible</th>
<th>Accessible by other VMs in OBPEE cluster</th>
</tr>
</thead>
<tbody>
<tr>
<td>22</td>
<td>SSH</td>
<td>N</td>
<td>N[1]</td>
<td>N</td>
</tr>
<tr>
<td>389</td>
<td>Local LDAP server for development purposes</td>
<td>N</td>
<td>N[1]</td>
<td>Y</td>
</tr>
<tr>
<td>443</td>
<td>Docker Registry</td>
<td>N</td>
<td>N</td>
<td>Y</td>
</tr>
<tr>
<td>636</td>
<td>Local LDAP server for development purposes</td>
<td>N</td>
<td>N</td>
<td>Y</td>
</tr>
<tr>
<td>2375</td>
<td>Docker Daemon</td>
<td>N</td>
<td>N</td>
<td>Y</td>
</tr>
<tr>
<td>2377</td>
<td>Docker Swarm</td>
<td>N</td>
<td>N</td>
<td>Y</td>
</tr>
<tr>
<td>7070</td>
<td>Control plane UI and REST (http–Application connector)</td>
<td>N</td>
<td>Y[2]</td>
<td>N</td>
</tr>
<tr>
<td>7443</td>
<td>Control plane UI and REST (https–Application connector)</td>
<td>N</td>
<td>Y[3]</td>
<td>N</td>
</tr>
<tr>
<td>7946</td>
<td>Docker Swarm</td>
<td>N</td>
<td>N</td>
<td>Y</td>
</tr>
<tr>
<td>8080</td>
<td>Component manager</td>
<td>N</td>
<td>N</td>
<td>Y</td>
</tr>
<tr>
<td>10000-10200</td>
<td>Ports assigned to the load balancer for the various instance containers such as peer, orderer, etc.</td>
<td>Y</td>
<td>Y[4]</td>
<td>Y</td>
</tr>
</tbody>
</table>

[1] SSH is only required if the VM console isn't accessible

[2] UI ports should be only accessible by machines needed for provisioning and configuring instances

[3] UI ports should be only accessible by machines needed for provisioning and configuring instances
[4] See the section on load balancer configuration

**Load Balancer Configuration**

If using an external load balancer, it will need to be configured to perform TLS termination and pass the ports as listed in the provisioning console to the appropriate VM host and port. Below is an example of the necessary port mappings as reported by the provisioning console:

**Figure 1-3  Load Balancer Port Mapping**

![Load Balancer Port Mapping](image)

See [Postrequisites When Using an External Load Balancer](#).

**Configuring Authentication and Authorization**

Authentication in Oracle Blockchain Platform is performed using an LDAP server. Users must have an account in LDAP in order to be able to use the service.

Users associated with certain LDAP groups are granted specific privileges as defined in [User Roles](#).

## Workflow for Administering Oracle Blockchain Platform

To start using Oracle Blockchain Platform, refer to the following tasks as a guide.

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>More Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prepare your hardware</td>
<td>Read through the suggested architectural designs and decide which is appropriate for your configuration. Ensure your hardware meets the required prerequisites.</td>
<td><a href="#">Supported Topologies</a> <a href="#">Prerequisites</a></td>
</tr>
</tbody>
</table>

---

 Oracle  

1-6
<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>More Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add and manage users and roles</td>
<td>A rudimentary LDAP server is provided with Oracle Blockchain Platform Enterprise Edition, however you'll need to use a third-party tool to add users and roles.</td>
<td>Configure an LDAP Server</td>
</tr>
<tr>
<td>Provision a service instance</td>
<td>Use the Create Instance wizard in Blockchain Platform Manager to create a service instance.</td>
<td>Provision an Instance using the Blockchain Platform Manager</td>
</tr>
<tr>
<td>Configure your blockchain network</td>
<td>Once your instance is created, you can use the Blockchain Platform Console to configure the network.</td>
<td>What’s the Console?</td>
</tr>
</tbody>
</table>

After you've created your instance and any required users, you can begin to use Oracle Blockchain Platform as described in *Using Oracle Blockchain Platform*. 
Design Your Oracle Blockchain Platform Configuration

Topics

- Prerequisites
- Supported Topologies

Prerequisites

Hardware

Your host environment should meet the following minimum requirements:

- 16 GB memory
- 500 GB available storage
- 4 CPUs

Virtual Machine Hosting Software

The following hypervisors are supported:

- Oracle VirtualBox
- VMWare Workstation
- VMWare ESXi

Additionally:

- VMs must be DNS-resolvable
- Date and time on the VMs and the client hosts for running a browser for Blockchain Platform Console and Blockchain Platform Manager must be synchronized.

Load Balancer

A lightweight load balancer is provided with Oracle Blockchain Platform for prototyping and development needs. It is not recommended for production use as it runs as part of the blockchain cluster.

An external load balancer capable of supporting TCP pass-through (and not just HTTP path mapping) can be provided, such as a dedicated NGINX 1.9.3+ server or F5.

LDAP

An LDAP server is provided for prototyping and development purposes, as managing blockchain instances and the blockchain network itself requires an identity management system.
This server isn't recommended for production use; we recommended you configure OpenLDAP 2.4.44+.

**Web Browsers**

All administrative tools included with Oracle Blockchain Platform can be accessed through these browsers:

- Mozilla Firefox
- Microsoft Edge
- Google Chrome
- Apple Safari

**Supported Topologies**

In addition to creating a topology in which both the founder and participant are on Oracle Blockchain Platform Enterprise Edition, the following interoperability scenarios are supported:

- Oracle Blockchain Platform Enterprise Edition Founder, Oracle Blockchain Platform Cloud Participant
- Oracle Blockchain Platform Enterprise Edition Founder, Hyperledger Fabric Participant
- Oracle Blockchain Platform Cloud Founder, Oracle Blockchain Platform Enterprise Edition participant
- Hyperledger Fabric founder, Oracle Blockchain Platform Enterprise Edition participant
3

Install Your Oracle Blockchain Platform Instance

Topics
• Deploy Your Virtual Machine
• Log on to Oracle Blockchain Platform for the First Time

Deploy Your Virtual Machine

Load Oracle Blockchain Platform Enterprise Edition on your Virtual Machine Hosting Software

1. Downloaded the blockchain package; it consists of an OVA image called obpee_19_3_2.ova.
2. Import the VM into your virtual machine hosting software. For example on Oracle VirtualBox:
   a. Select File then Import Appliance, and browse to the directory where the OVA has been extracted.
   b. On the Appliance settings page, you may check Reinitialize the MAC address of all network cards if you plan on running more than 1 VM in your setup. You can create multiple VMs by either importing the appliance multiple times, or cloning the VM immediately after it's imported. Click Import.
3. After a few minutes, the VM will be displayed in the list of machines in VirtualBox Manager. Right-click on the VM and select Settings:
   a. Under System on the Motherboard tab, select Hardware Clock in UTC Time to ensure the guest VM and the host's clocks are consistent in terms of timezones.
   b. Under Network on the Adapter tab connected to the network you want the VM to be on, ensure Enable Network Adapter is selected, and select Bridged Adapter.

The VM is now ready to be used.

Expand the Available Space for the Ledger

Because the ledger and logs for Oracle Blockchain Platform are persistent, you need to expand the root volume to ensure you don't run out of space.

1. Add a hard disk for the SCSI controller. In VirtualBox click Settings then select Storage. Add a new hard disk. The recommended file type is VHD (Virtual Hard Disk); select either fixed or extensible mode.
2. Create the logical volume partition: sudo fdisk -c -u /dev/sdb
Enter the following subcommands sequentially:

n Create new partition Press 
p Choose primary partition use p 
1 Choose 1 for the primary partition 
after this press return key twice for max allocation 
t Change the type 
e Change the partition type to Linux LVM 
p Print the partition 
w write the changes

3. **Create the new physical volume:** `sudo pvcreate /dev/sdb1`

4. **Restart the system.**

5. **Verify the physical volume:**

   ```
sudo pvs
PV VG Fmt Attr PSize PFree
/dev/sda2 vg00 lvm2 a-- 96.66g <25.41g
/dev/sdb1 lvm2 i <200.00g <200.00g
```

6. **Add** `/dev/sdb1` **to vg00 to extend the size to get more space for expanding the logical volume:** `sudo vgextend vg00 /dev/sdb1`

7. **Check the size of the volume group:**

   ```
sudo vgs
VG #PV #LV #SN Attr VSize VFree
vg00 2 5 0 wz-n <296.66g 225.40g
```

   *In this example, 225.4 GB are free.*

8. **After extending, resize the file system:** `sudo resize2fs /dev/vg00/root`

9. **Check the logical volume:**

   ```
   /dev/vg00/root
df -h
   /dev/mapper/vg00-root 194G 18G 169G 10% /
   ```

**Start Oracle Blockchain Platform**

Once your network and system settings are configured, start the VM by selecting it and clicking **Start**. It should start within a minute.

---

**Log on to Oracle Blockchain Platform for the First Time**

After you've deployed and started Oracle Blockchain Platform Enterprise Edition on your VM hosting software, you can log on to Blockchain Platform Manager to create an instance.

You can directly log on to the Platform Manager by using the URL:

```https://<hostname of your VM>:7443/console/index.html```
The initial user name is `obpadmin` and the password is `welcome1`. This user is only meant for performing initial configuration and does not have instance creation privileges.

In order to use the internal LDAP server, the `admin` password must first be changed. Blockchain Platform Manager will not allow you to use an internal LDAP configuration with the default password.

**Set the Blockchain Platform Manager Name**

On the **Configuration** page of Blockchain Platform Manager, you can set a name for the Platform Manager.

Once the name for the Platform Manager has been set, any users added to the LDAP server will be associated with this name. If you change the name after adding users, those users will lose access to Blockchain Platform Manager and any Oracle Blockchain Platform instances.
User Management

Topics

- Configure an LDAP Server
- Add Users to Your LDAP Server
- User Roles

Configure an LDAP Server

An LDAP server is included with Oracle Blockchain Platform Enterprise Edition or you can integrate your own.

Each instance within a Blockchain Platform Manager uses the same LDAP server. You can create multiple Blockchain Platform Manager instances, and each one can use a different LDAP server or share an LDAP server.

Lifecycle of Identity Resources within Oracle Blockchain Platform

When you provision an instance through Blockchain Platform Manager, it deploys the embedded LDAP server (if you're not providing your own), creates the LDAP groups OBP_<platform-name>_<instance-name>_xxx.

When you delete an instance, Blockchain Platform Manager removes all the LDAP assets such as the LDAP groups from an LDAP server you have provided.

- Configure the Built-In LDAP Server
- Configure an External LDAP Server

Configure the Built-In LDAP Server

The built-in LDAP server has a default configuration already set up when you log in. You can use it for testing, or modify the configuration to meet your needs.

1. Open the Configuration tab.
2. Click Add New.
3. Enter the configuration information for the LDAP server:
   a. Configuration Name:
      Name must contain only ASCII alphanumerics and underscores.
   b. LDAP Server Type:
      Only OpenLDAP is supported at this time.
   c. Host:
      Enter the fully-qualified host name of the directory server.
d. Port:
Enter the port number of the directory server.

e. TLS Enabled:
Setting this to True means you will connect to the directory server using a user name and password via SSL.

f. Connect Timeout:
In milliseconds.

g. Base DN:
Enter the base distinguished name of the directory you want to connect to. It should be in the form: `ou=organizationunit,dc=mycompany,dc=com`

h. TLS Certificate Archive for LDAP Client:
If you’re using a third-party TLS certificate or self-signed certificate, upload it.

i. Bind User DN:
The distinguished name of your administrative user account.

j. Bind User Password:
The password for the account.

k. UserName Attribute:
This is the filter used when searching to convert a login user name to a distinguished name.

l. User Class Name:
The attribute value to a user object in the directory.

m. GroupName Attribute:
This is the filter used when searching to convert a group name to a distinguished name.

n. Group Membership Attribute:
The membership attribute name of the group.

o. Group Class Name:
The ObjectClass attribute value for a group object in the directory.

4. Click **Test Configuration** to ensure your settings work. The test results show if the configuration was successful.

5. Click **Save**. Your configuration is now available to be used by any instances you provision.

Once you've selected your LDAP configuration by selecting it in the **Active LDAP Configuration** field, you need to log out of Blockchain Platform Manager with your administrative ID, and log in with a user ID that exists in the LDAP server as described in **Add Users to Your LDAP Server**.
Configure an External LDAP Server

If you don't want to use the LDAP server provided with the product, you must have installed your own before completing this configuration step.

An external LDAP server should be installed for any production environment. It should be protected by TLS certificates - self-signed certificates should be used for internal testing only. If you are using self-signed certificates, complete these steps before configuring the LDAP server through Blockchain Platform Manager:

1. Generate a root CA key/certificate pair.
2. Generate a server key/certificate pair signed using the root CA pair.

When configuring the server in Blockchain Platform Manager you will need to upload the root CA certificate.

1. Open the **Configuration** tab.
2. Click **Add New**.
3. Enter the configuration information for the LDAP server:
   a. **Configuration Name:**
      Name must contain only ASCII alphanumerics and underscores.
   b. **LDAP Server Type:**
      Only OpenLDAP is supported at this time.
   c. **Host:**
      Enter the fully-qualified host name of the directory server.
   d. **Port:**
      Enter the port number of the directory server.
   e. **TLS Enabled:**
      Setting this to True means you will connect to the directory server using a user name and password via SSL.
   f. **Connect Timeout:**
      In milliseconds.
   g. **Base DN:**
      Enter the base distinguished name of the directory you want to connect to. It should be in the form: `ou=organizationunit,dc=mycompany,dc=com`
   h. **TLS Certificate Archive for LDAP Client:**
      If you’re using a third-party TLS certificate or self-signed certificate, upload it.
   i. **Bind User DN:**
      The distinguished name of your administrative user account.
   j. **Bind User Password:**
      The password for the account.
   k. **UserName Attribute:**
This is the filter used when searching to convert a login user name to a distinguished name.

1. **User Class Name:**
   The attribute value to a user object in the directory.

2. **GroupName Attribute:**
   This is the filter used when searching to convert a group name to a distinguished name.

3. **Group Membership Attribute:**
   The membership attribute name of the group.

4. **Group Class Name:**
   The ObjectClass attribute value for a group object in the directory.

4. **Click Test Configuration** to ensure your settings work. The test results show if the configuration was successful.

5. **Click Save.** Your configuration is now available to be used by any instances you provision.

Once you've selected your LDAP configuration by selecting it in the **Active LDAP Configuration** field, you need to log out of Blockchain Platform Manager with your administrative ID, and log in with a user ID that exists in the LDAP server.

### Add Users to Your LDAP Server

Once you've configured your LDAP server in Blockchain Platform Manager, you need to add users to the LDAP server to create an instance.

The following steps describe how to add the initial user to the built-in LDAP server:

1. **Log into the VM instance as a Unix user.** The initial user name and password are `oracle` and `Welcome1`. You'll be prompted to change the password immediately.

2. **Change directories to** `/u01/blockchain/ldap/environment` **and run the adduser.sh script:**
   - `cd /u01/blockchain/ldap/environment/`
   - `./adduser.sh user_name platform_name`
   - **You will be prompted to enter a password for the new user, as well as a password for the administrator who will authenticate user and group addition requests.**
   - **The script will add a new user to the group OBP_<platform name>_CP_ADMIN which will have administrative access to Blockchain Platform Manager in order to create and modify instances.**

The following steps describe how to add the initial user to your separately-installed LDAP server:

1. **Create your administrative user if one doesn't already exist.**

2. **Create the OBP_<platform name>_CP_ADMIN group if it doesn't exist.**

3. **Add the user as a member of the OBP_<platform name>_CP_ADMIN group.**
Ensure that you've logged out of Blockchain Platform Manager, and then log in using this user ID and password. You can now provision a Oracle Blockchain Platform instance.

**User Roles**

This overview describes the roles that are relevant to Oracle Blockchain Platform. Anyone who uses or administers Oracle Blockchain Platform must be added to the LDAP server and granted the correct user role.

Below are the roles that are available for Oracle Blockchain Platform.

<table>
<thead>
<tr>
<th>User Role</th>
<th>Role Name in LDAP</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application</td>
<td>LDAP group OBP_&lt;platform-name&gt;_&lt;instance-name&gt;</td>
<td>Security identifier for an individual instance.</td>
</tr>
<tr>
<td>CA Administrator</td>
<td>LDAP group OBP_&lt;platform-name&gt;_&lt;instance-name&gt;_CA_ADMIN</td>
<td>The CA Admin role is the bootstrap and overall administrator for the Oracle Blockchain Platform application.</td>
</tr>
<tr>
<td>Instance Administrator</td>
<td>LDAP group OBP_&lt;platform-name&gt;_&lt;instance-name&gt;_ADMIN</td>
<td>User with this role can manage instance via console UI or REST. See the table in Access Control List for Console Function by User Roles for a complete list of console functions available for this user role.</td>
</tr>
<tr>
<td>Instance User</td>
<td>LDAP group OBP_&lt;platform-name&gt;_&lt;instance-name&gt;_USER</td>
<td>User with this role can view instance via console UI or REST. See the table in Access Control List for Console Function by User Roles for a complete list of console functions available for this user role.</td>
</tr>
<tr>
<td>REST Proxy Client</td>
<td>LDAP group OBP_&lt;platform-name&gt;_&lt;instance-name&gt;_REST</td>
<td>User with this role can call REST proxy to execute transactions using the default enrollment.</td>
</tr>
<tr>
<td>Custom REST Client</td>
<td>LDAP group OBP_&lt;platform-name&gt;_&lt;instance-name&gt;<em>REST</em>&lt;custom-enrollment&gt;</td>
<td>User with this role can call REST proxy to execute transactions using a custom enrollment.</td>
</tr>
<tr>
<td>Control Plane management</td>
<td>LDAP group OBP_&lt;platform-name&gt;_&lt;instance-name&gt;_CP_ADMIN</td>
<td>User can provision a new Oracle Blockchain Platform instance, configure existing instances, set the LDAP configuration, and perform life cycle operations on Oracle Blockchain Platform instances.</td>
</tr>
</tbody>
</table>

**Access Control List for Console Function by User Roles**

The following table lists which console features are available to the Instance Administrator and Instance User roles.
<table>
<thead>
<tr>
<th>Feature</th>
<th>Instance Administrator</th>
<th>Instance User</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dashboard</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Network: list orgs</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Network: add orgs</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Network: Ordering service setting</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Network: Export certificates</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Network: Export orderer settings</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Node: list</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Node: start/stop/restart</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Node: view attributes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Node: edit attributes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Node: view metrics</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Node: Export/Import Peers</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Peer Node: list channels</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Peer Node: join channel</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Peer Node: list chaincode</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Channel: list</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Channel: create</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Channel: add org to channel</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Channel: Update ordering service settings</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Channel: view/query ledger</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Channel: list instantiated chaincode</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Channel: list joined peers</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Channel: set anchor peer</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Channel: upgrade chaincode</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Chaincode: list</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Chaincode: install</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Chaincode: instantiate</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Sample chaincode: install</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Sample chaincode: instantiate</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Sample chaincode: invoke</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>CRL</td>
<td>Yes</td>
<td>No</td>
</tr>
</tbody>
</table>
Provision an Instance

Topics

- Before You Create an Oracle Blockchain Platform Instance
- Provision an Instance using the Blockchain Platform Manager
- Provision an Instance Using REST APIs
- Postrequisites When Using an External Load Balancer

Before You Create an Oracle Blockchain Platform Instance

Before you provision Oracle Blockchain Platform, decide if a developer or enterprise instance meets your needs.

Deciding Which Provisioning Shape to Use

When provisioning an instance, you choose between two configurations. Migration between these options isn't supported currently.

<table>
<thead>
<tr>
<th>Configuration</th>
<th>Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>Developer</td>
<td>Default configuration is a single platform VM running all other blockchain functions such as peers, orderers, CAs, console, REST proxy and an internal load balancer, and 1 chaincode runtime container VM for running chaincode</td>
</tr>
<tr>
<td></td>
<td>• 2 Fabric-CA nodes</td>
</tr>
<tr>
<td></td>
<td>• 3-node single VM Kafka/Zookeeper cluster (Founder only)</td>
</tr>
<tr>
<td></td>
<td>• Up to 14 Peer nodes</td>
</tr>
<tr>
<td></td>
<td>• Dynamically managed chaincode execution containers</td>
</tr>
<tr>
<td></td>
<td>• Console service for operations web user interface</td>
</tr>
<tr>
<td></td>
<td>• REST proxy service for RESTful API</td>
</tr>
<tr>
<td></td>
<td>• LDAP server integration for authentication and role management</td>
</tr>
<tr>
<td></td>
<td>• Load balancer</td>
</tr>
<tr>
<td>Enterprise</td>
<td>Default configuration is 3 platform VMs running all other blockchain functions such as peers, orderers, CAs, console, REST proxy and an internal load balancer, and 1 chaincode runtime container VM for running chaincode</td>
</tr>
<tr>
<td></td>
<td>• 2 Fabric-CA nodes</td>
</tr>
<tr>
<td></td>
<td>• 3-VM Kafka/Zookeeper cluster (Founder only) for high availability</td>
</tr>
<tr>
<td></td>
<td>• Up to 14 Peer nodes spread across separate virtual machines</td>
</tr>
<tr>
<td></td>
<td>• Dynamically managed chaincode execution containers in an isolated virtual machine</td>
</tr>
<tr>
<td></td>
<td>• Console service for operations web user interface replicated across separate virtual machines for high availability</td>
</tr>
<tr>
<td></td>
<td>• REST proxy service for RESTful API</td>
</tr>
<tr>
<td></td>
<td>• LDAP server integration for authentication and role management</td>
</tr>
<tr>
<td></td>
<td>• Load balancer</td>
</tr>
</tbody>
</table>
Provision an Instance using the Blockchain Platform Manager

To create a blockchain founder or participant instance in Blockchain Platform Manager, use the Create New Instance wizard.

There are two types of Oracle Blockchain Platform instances you can provision:

- **Founder organization**: a complete blockchain environment, including a new network to which participants can join later on.
- **Participant instance**: if there is already a founder organization you want to join, you can create a participant instance if your credentials provide you with access to the network.

1. In Blockchain Platform Manager, open the Instances page.
2. Select **Create Instance**.
3. Complete the following fields:

<table>
<thead>
<tr>
<th>Section</th>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>General</td>
<td><strong>Instance Name</strong></td>
<td>Enter a name for your Oracle Blockchain Platform instance. The instance name:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Must contain one or more characters.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Must not exceed 15 characters.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Must start with an ASCII letter: a to z.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Must contain only ASCII lower-case letters or numbers.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Must not contain a hyphen.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Must not contain any other special characters.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Must be unique within the identity domain.</td>
</tr>
<tr>
<td></td>
<td><strong>Description</strong></td>
<td>Optional. Enter a short description of the Oracle Blockchain Platform instance.</td>
</tr>
<tr>
<td>Role</td>
<td><strong>Role</strong></td>
<td>Select Founder to create a complete blockchain environment. This instance becomes the founder organization and you can onboard new participants in the network later. Select Participant to create an instance that will join an existing blockchain network created elsewhere before this instance can be used.</td>
</tr>
<tr>
<td>Configuration</td>
<td><strong>Configuration</strong></td>
<td>Select a provisioning shape which meets the needs of your deployment:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>Developer</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>Enterprise</strong></td>
</tr>
<tr>
<td>Peers</td>
<td><strong>Peers</strong></td>
<td>Specify the number of peer nodes to be initially created in this service instance. You can create between 1 and 14 peer nodes. You can create additional peer nodes in the Oracle Blockchain Platform console at a later time.</td>
</tr>
<tr>
<td>Cluster Configuration</td>
<td><strong>Platform Host</strong></td>
<td>Add the fully qualified host name of the VM hosting Oracle Blockchain Platform. For Developer instances you need to provide one VM. For Enterprise instances you need to provide three VMs to create a high-availability cluster.</td>
</tr>
<tr>
<td></td>
<td><strong>Chaincodes Host</strong></td>
<td>Add the fully qualified host name of the VM hosting the chaincodes.</td>
</tr>
</tbody>
</table>
### Section | Field | Description
---|---|---
**Zookeeper/Kafka Host** | | Add the fully qualified names of the VMs hosting the Zookeeper/Kafka orderer cluster (the platform hosts). Developer instances will have 1 VM, Enterprise will have 3 VMs to create a high-availability cluster.

**Additional Configuration** | **Use External Load Balancer** | Select if you want to use an external load balancer instead of the one provided by Oracle Blockchain Platform Enterprise Edition. Enter the fully qualified domain name and port of the load balancer. Upload the TLS root CA certificate. The TLS root CA certificate must be named rootCA.zip and contain a single file named tls-ca.pem.

**Enable TLS for Default Load Balancer** | | If you want to use the load balancer provided by Oracle Blockchain Platform Enterprise Edition, select this option.

**Third Party CA Archive** | | Optional. Oracle Blockchain Platform includes a certificate authority (CA), which is used to create self-signed certificates for all blockchain nodes in your instance. If you want to use certificates from your own certificate authority and use the Oracle Blockchain Platform certificate authority as an intermediary CA, you can upload your CA archive. The certificate you upload will be used to sign the intermediary certificates for Oracle Blockchain Platform nodes, thus including them under your root CA chain.

The archive is a zip file which contains the following files:

- **CA chain** - named xxxca-chain.pem. The entire CA file sequence from the signing CA to the top-level CA should be present.
- **key** - named xxxca-key.pem. The key should be a 256-bit elliptic curve key. The prime256v1 curve is recommended. The key should be an unencrypted private key in PKCS #8 format.
- **certificate** - named xxxca-cert.pem. Must be in Base64 format. Must include the Subject Key Identifier extension. where xxx is an identifier of your choice. The archive must be less than 2MB.

4. Verify that the details are correct, and click Confirm.

Once your instance has been created and is listed in the Instances list, you can launch the service console from the menu next to the instance name. Use the console to configure your network as described in *Using Oracle Blockchain Platform*.

### Provision an Instance Using REST APIs

You can provision an Oracle Blockchain Platform instance using a REST API.

The following example shows how to create an Oracle Blockchain Platform instance using REST API:

```bash
curl -X POST \
-u <username>:<password> \nhttp://localhost:7070/api/v1/blockchainPlatforms/instances \
-H "Content-Type: multipart/form-data; boundary=----WebKitFormBoundary7MA4YWxkTrZu0gW©"
```

---

**Chapter 5**

**Provision an Instance Using REST APIs**

You can provision an Oracle Blockchain Platform instance using a REST API.

The following example shows how to create an Oracle Blockchain Platform instance using REST API:

```bash
curl -X POST \
-u <username>:<password> \nhttp://localhost:7070/api/v1/blockchainPlatforms/instances \
-H "Content-Type: multipart/form-data; boundary=----WebKitFormBoundary7MA4YWxkTrZu0gW©"
```
-F 'payload={
  "name": "obpinstance1",
  "desc": "test instance",
  "platformRole": "founder",
  "configuration": "Developer",
  "peer": 4,
  "cluster": {
    "platformHosts": [
      "10.182.73.23",
      "10.182.73.20"
    ],
    "crcHosts": [
      "10.182.73.23",
      "10.182.73.20"
    ]
  },
  "additionalConfiguration": {
    "instanceFQDN": "domain.host.com"
  }
}'

• **name**
  - Must contain one or more characters.
  - Must not exceed 15 characters.
  - Must start with an ASCII letter: a to z.
  - Must contain only ASCII lower-case letters or numbers.
  - Must not contain a hyphen.
  - Must not contain any other special characters.
  - Must be unique within the identity domain.

• **desc**
  - Optional: Enter a description of the instance

• **platformRole**
  - Must be set to developer or founder

• **configuration**
  - **Developer**: A 1 Kafka orderer and 3 OCPU total in 1 VM
  - **Enterprise**: A 3 node Kafka cluster and 3 X VM

• **peer**
  - Specify the number of peer nodes that will be initially created in this service instance.
  - 1 to 14 peer nodes can be created.

• **cluster**
  - Enter the information for your cluster:
    * **platformHosts**: the machines hosting your platform cluster
    * **crcHosts**: the machines hosting the Kafka/Zookeeper cluster
• **instanceFQDN**
  - The fully qualified domain name of your external load balancer. This is used exclusively for external load balancers - if you're not using an external load balancer, you don't need to specify this parameter.

## Postrequisites When Using an External Load Balancer

When provisioning your instance, if you are using an external load balancer you must have selected this during the provisioning steps and uploaded the TLS root CA certificate as described in Provision an Instance using the Blockchain Platform Manager or Provision an Instance Using REST APIs.

Once this is done you can configure your load balancer. The blockchain instance will be listening on a variety of ports which will need mapped to external ports. The ports used will vary depending on the configuration; namely the amount of peers.

1. Obtain the complete list of ports needing mapping for your instance. Open the Instance Details page for your instance on Blockchain Platform Manager, then click **LBR Port Map**. Record the ports listed.

2. In your load balancer, do a mapping as shown in the Nginx syntax example below, where *my.blockchain.example.com* is the FQDN of the blockchain instance (internal side):

   ```
   ...stream {
      upstream port1 {
         server my.blockchain.example.com:10001;
      }
      server {
         listen *:10003 ssl;
         ssl_certificate /etc/nginx/server.pem; # use your own certificate/key
         ssl_certificate_key /etc/nginx/serverkey.pem;
         proxy_pass port1;
      }
   ...}
   
   3. Repeat for every port listed in the port map.

   ✍️ **Note:**

   If at some point in the future you scale out your instance by adding new peers, remember to map those new peers using the steps above.
Manage Oracle Blockchain Platform

Once you've provisioned your instance, you can manage it in Blockchain Platform Manager.

Topics

• View Instance Details
• Start or Stop an Instance
• Delete an Instance
• Scale an Instance In or Out
• Patch an Instance

View Instance Details

Clicking on your instance name in Blockchain Platform Manager opens the Instances tab displaying details about the instance.

The Instance Details page lists information such as the location of the logs and ledger, as well as the health of the instance. You can also see all the hosts and their status.

The Patching page lists all the patches that have been applied to the instance, as well as any available patches that haven't been applied.

You can manage the instance from the Actions menu, or launch the Oracle Blockchain Platform Service Console to manage your blockchain network.

Open the Configuration tab to access the LDAP and platform configuration tabs. You can update your LDAP or platform configuration if needed. You may want to disable the default user.

Start or Stop an Instance

You can start or stop an instance in the Blockchain Platform Manager.

To start or stop an instance:

1. In Blockchain Platform Manager, find your instance and select the menu beside it.
2. Select Start or Stop. You'll be prompted to confirm your selection.

Delete an Instance

You can delete your instance in Blockchain Platform Manager.

To delete your instance:

1. Open Blockchain Platform Manager and find your instance.
2. From the menu beside your instance, select **Terminate**.

3. You'll be prompted to confirm you action. Click **Confirm**.

## Scale an Instance In or Out

You can scale an instance in or out in Blockchain Platform Manager.

### Scale Out

You can scale out your instance by creating new VMs, replicas, or peers:

1. In Blockchain Platform Manager open the menu beside your instance name and click **Scale Out**.

2. You can scale out using any of these methods:
   - **New VMs**: adds a new VM to the cluster; platform host, chaincode host, or Zookeeper/Kafka host.
   - **New Replicas**: adds additional nodes; REST proxy or CA.
   - **New Peers**: adds additional peers.

### Scale In

You can scale in your instance by deleting peers.

Before scaling in an instance, you should transfer all this peer's responsibilities to other running peers, and then remove all the responsibilities this peer has.

- Check all other peers' gossip bootstrap address lists, remove the peer address, and add another running peer's address if needed. After peer configuration change, restart the peer.
- Check all channels' anchor peer lists, remove the peer from the anchor peer lists, and add another running peer to the anchor peer list if needed.
- If a channel or chaincode is only joined or instantiated in this peer, you should consider using another running peer to join the same channel and instantiate the same chaincode.

1. In Blockchain Platform Manager open the menu beside your instance name and click **Scale In**.

2. Enter the hostname or IP address of the peer you want to delete. To delete more than one peer, click Add Peer and enter the information for the additional peer.

## Patch an Instance

You can patch your instance in Blockchain Platform Manager.

A patch package includes:

- A metadata file
- One or more scripts to be run during patching
- Docker images for different components

Blockchain Platform Manager is treated as a component, and its patch is included in the same package together with other components.
The patch package is a rolling patch; a newer patch is always a superset of an older patch within the same release.

1. Download any desired patches from support.oracle.com.
2. Open the Patches tab.
3. Click Register Patch and select the patch you've downloaded.
4. Select your instance to open the Instance Details page and select the Patches tab.
5. Select the registered patch you want to apply.

Blockchain Platform Manager extracts the patch package, pushes the component docker images to the docker registry, and stores the metadata file and scripts in local database.
Monitor and Troubleshoot Your Instance

Topics

- Logging

Logging

After Oracle Blockchain Platform has started successfully, a container log will be written to `/u01/obp-logs` when its size reaches 10MB. You can check the current logs in the docker containers using docker commands.

Upon failure of provisioning, in most cases you would want to use a docker command to check the container status and logs, such as `docker ps`, `docker inspect`, `docker service ls`, and `docker service ps`. Logs will not be found under `/u01/obp-logs` because either the container creation failed in the middle of the docker daemon preparing the container, control has not been transferred to `ENTRYPOINT` command, or the container has exited due to error condition before the log reached 10MB.

For a Blockchain instance, for the nodes including peer, orderer, CA, and REST Proxy, the log level can be updated through the console.

For component manager and node manager on the VM, you can use the environment variable `BCS_LOG_LEVEL` to set the log level.

To set the log level for component manager:

1. Log in to your virtual machine
2. Open `/home/oracle/startcm.sh` in an editor.
3. Add the following line next to the line starting with `docker run`:

   ```sh
   -e "BCS_LOG_LEVEL=DEBUG" \
   ```

   where the supported log levels are `ERROR`, `WARNING`, `INFO`, and `DEBUG`.
4. Restart the component manager:

   ```sh
   sudo systemctl restart compmanager.service
   ```

To set the log level for node manager:

1. Find your node manager service name:

   ```sh
   docker service ls
   ```
2. Use the following Docker command:

   ```sh
   docker service update --env-add BCS_LOG_LEVEL=DEBUG <NM_service_name>
   ```
where the supported log levels are ERROR, WARNING, INFO, and DEBUG.
Accessibility Features and Tips for Oracle Blockchain Platform

This topic describes accessibility features and information for Oracle Blockchain Platform.

Table A-1  Keyboard Shortcuts for Blockchain Platform Manager

<table>
<thead>
<tr>
<th>Task</th>
<th>Keyboard Shortcut</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create an Oracle Blockchain Platform instance.</td>
<td>Alt+Shift+C</td>
</tr>
<tr>
<td>Terminate an Oracle Blockchain Platform instance</td>
<td>Alt+Shift+T</td>
</tr>
<tr>
<td>Navigate to the Patch page</td>
<td>Alt+Shift+P</td>
</tr>
<tr>
<td>Refresh the Instance Summary page</td>
<td>Alt+Shift+R</td>
</tr>
</tbody>
</table>