Oracle® Communications Security Shield Cloud Service

Installation and Maintenance Guide

F24352-24
February 2024

Oracle Communications Security Shield Cloud Service Installation and Maintenance Guide,

F24352-24

Contents

About This Guide

The Security Shield Installation and Maintenance Guide provides information about the components of the Oracle® Communications Security Shield Cloud Service (Security Shield) that you need to install and manage. The guide explains each component along with the high-level installation process and detailed installation instructions. The guide also explains the related maintenance and management operations.

Security Shield Operations Documented in this Guide

- Establishing an Oracle Cloud account and obtaining the Security Shield subscription
- Downloading the Security Shield software
- Installing and managing the Cloud Communications Service
- Configuring Security Shield on the Oracle Communications Session Border Controller

Documentation Set

The following table describes the documents included in the Oracle® Communications Security Shield Cloud Service (Security Shield) documentation set.

Security Shield Installation and Maintenance Guide	Contains conceptual and procedural information for installing and maintaining the Security Shield.
Security Shield Security and Privacy Guide	Contains conceptual and procedural information for securing the Security Shield operations.
Security Shield User's Guide	Contains the product overview along with conceptual and procedural information about using the Security Shield Dashboard.
Security Shield What's New	Contains information about the release including new features, caveats, known issues, and limitations.

Related Documentation

The following list describes related documentation for the Oracle® Communications Security Shield Cloud Service (Security Shield). You can find the listed documents on http://docs.oracle.com/en/industries/communications/ in the "Session Border Controller Documentation" section.

ACLI Configuration Guide	Contains information about the administration and software configuration of the Oracle Communications Session Border Controller.

ACLI Reference Guide	Contains explanations of how to use the ACLI, as an alphabetical listings and descriptions of all ACLI commands and configuration parameters.
Call Traffic Monitoring Guide	Contains information about traffic monitoring and packet traces as collected on the system. This guide also includes Web GUI configuration used for the SIP Monitor and Trace application.
Installation and Platform Preparation Guide	Contains conceptual and procedural information for system provisioning, software installations, and upgrades.
Maintenance and Troubleshooting Guide	Contains information about Oracle Communications Session Border Controller logs, performance announcements, system management, inventory management, upgrades, working with configurations, and managing backups and archives.

History

The following table provides the revision history for this document. Oracle updates the documentation set with each software release. When one or more of the documents requires an update between software releases, Oracle issues an update limited to the affected documents.

Dates	Release Numbers and Revisions
June 2020	20.0.0.0.0
July 2020	20.1.0.0.0
September 2020	Documentation Update • Adds the Note about using Sp Ls to the "Add the Security Shield SPL Plug-in" topic.
October 2020	20.2.0.0.0
October 2020	Documentation Update • Updates the SPL entry for ocss-client-config=httpClientConfigName in the "Configure the Oracle Session Border Controller for the Security Shield Service" topic.
December 2020	Documentation Update • Clarifies the Operating System requirement for the Policy Decision Engine and Cloud Communication Service.
February 2021	20.3.0.0.0
March 2021	Documentation Update • Adds a cross reference to the "CCS Configuration Behind NAT or a Firewall" topic in the *Security Shield Security and Privacy* Guide from the "Install, Configure, and Activate the Cloud Communication Service" topic in this guide.

Dates	Release Numbers and Revisions
April 2021	Documentation Update • Adds the Note that the Cloud Communication Service (CCS) does not support simultaneous use of the same CCS instance to the "Install, Configure, and Activate the Cloud Communication Service" topic.
June 2021	21.0.0.0.0
August 2021	21.1.0.0.0
November 2021	21.2.0.0.0
February 2022	21.3.0.0.0
May 2022	22.0.0.0.0
August 2022	22.1.0.0.0
September 2022	• Updates the procedure in the "Add the Security Shield SPL Plug-in" topic.
February 2023	23.0.0.0.0
May 2023	• 23.1.0.0.0 • Updates the list of trusted Certificate Authorities in Appendix C.
August 2023	23.2.0.0.0
September 2023	Documentation update • Updates the "Install, Configure, and Activate the Cloud Communication Service", "Deactivate the Cloud Communication Service", and "Update the Cloud Communication Service" topics to exclude Podman.
November 2023	23.3.0.0.0
December 2023	23.3.0.0.0 Updates the following topics to include Podman for Oracle Linux 8: • Install, Configure, and Activate the Cloud Communication Service • Deactivate the Cloud Communication Service • Update the Cloud Communication Service
February 2024	24.0.0.0.0

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.

1 Summary of the Oracle® Communications Security Shield Cloud Service

Oracle® Communications Security Shield Cloud Service (Security Shield) is subscription-based Oracle® Cloud service that provides the following services, components, and operations to help secure your telephony network.

The Security Shield service evaluates SIP-based calls that it receives from the Oracle Communications Session Border Controller (OCSBC), determines a reputation score for each call, and sends the corresponding action to the OCSBC to enforce.

The Security Shield service provides a Dashboard where you can view data about call traffic and manage how you want the SBC to handle risky calls. Through the tabs on the Dashboard, you can set thresholds for call activity and you can create lists of phone numbers for the Security Shield service to use when determining the reputation scores and actions.

Most of the Security Shield service components and operations reside in the Oracle Cloud, but you must install the Cloud Communications Service (CCS) on-premises for communication between your Session Border Controller and the Oracle Cloud. You must also enable your Session Border Controller for Security Shield operations. The Security Shield service in the Oracle Cloud hosts the Dashboard and the Analytics Cloud Service.

- The Dashboard is the Graphical User Interface that displays visualizations of call activity and other Security Shield operations.
- The Security Shield service applies rules, provides call behavior analytics, sends the enforcement actions to the Session Border Controller.
- The Analytics Cloud Service uses a combination of multiple services designed to perform advanced analytics on a sample data set.
- The Cloud Communication Service (CCS) establishes a secure ground-to-cloud tunnel for on-premises Security Shield components to communicate with Security Shield components in the Oracle Cloud.

You must contact your Oracle sales representative to subscribe to the Security Shield because it is not available for purchase through the Oracle online marketplace.

See the *Security Shield User's Guide* for overviews, illustrations, and descriptions of the service. Oracle Recommends that you read this guide first.

See the *Security Shield Installation and Maintenance Guide* for information about obtaining the service as well as deployment instructions for installing the on-premises components.

Topics:

2

Security Shield Deployment Overview

The Oracle Communications Security Shield (Security Shield) consists of several components that interact with one other to provide the service. Major components include the Oracle Communications Session Border Controller (OCSBC), (Security Shield) services, and the Cloud Communication Service (CCS). The Security Shield service resides in the Oracle Cloud, while the OCSBC and the CCS reside on premises. You must download and install the CCS software and Security Shield SPL plug-in, as well as configure the OCSBC for Security Shield operations. See the Security Shield User's Guide to learn about the components and their respective operations.

Topics:

Version Requirements for External Components

If you use any of the following components that are external to the Oracle Cloud, the Oracle® Communications Security Shield Cloud Service (Security Shield) requires the following versions at the minimum.

Data Collection and Enforcement Points

Session Border Controller

- Use Oracle Communications Session Border Controller release S-Cz8.4.0 p10 or higher.

> **Note:**
>
> If you require performance higher than 50 calls per second, Oracle recommends that you use version S-Cz9.0 or higher with the latest patch.

- SPL package—Customers currently using Security Shield must upgrade their Session Border Controllers to the latest released SPL, but only after upgrading their tenant to the latest Security Shield release. Get the latest version available for download from Oracle Software Delivery Cloud or My Oracle Support. Install the SPL on the external-facing realm.
- SPL Engine—C3.1.14 or higher
- Virtualization—Run as a Virtual Network Function (VNF).
- See the *Oracle Communications Session Border Controller Platform and Installation Guide* for information about platform support.

Session Router

- Use S-Cz 9.0.0 or higher.
- 4 vCPU cores, 32 GB RAM-Hardware, 8GB RAM-Virtual Machines, 20GB HDD, and 8 vNICs
- Install the Security Shield SPL on the external facing realm.
- Session Router runs in the session stateful configuration mode, only. Security Shield does not support other configurations and modes.
- SPL Engine—C3.1.14 or higher
- Virtualization—Run as a Virtual Network Function (VNF).
- See the *Oracle Session Border Controller ACLI Configuration Guide*, Oracle Communications Session Router, and Session Router Data Sheet documentation.

Service Authentication and Connectivity

Oracle Cloud Communication Service

- CCS—Use the latest version available for download from Oracle Software Delivery Cloud or My Oracle Support.
- Hardware—Oracle X8-2 or equivalent
- CPUs—1
- Memory—250MB
- Disk—100MB
- Throughput—1GBps NIC capacity
- Docker—Use version 18.09.1 or higher with the daemon running as a service
- Oracle Linux—Use version 7.6 or higher, or equivalent with compatible Red Hat Compatible Kernel (same version).

Podman

- Version 4.4.1 or higher

Browser Support

Oracle recommends that you use the latest versions of Google Chrome, Mozilla Firefox, and Microsoft Edge as of the date of this release, for the best user experience.

Oracle does not support Internet Explorer 11.

> **Note:**
>
> After upgrading the software, clear the browser cache before using the Web GUI.

Cloud Communications Service Deployment, Management, and Work Flow

When you deploy the Cloud Communications Service (CCS), you must run the supplied scripts to install, configure, activate, deactivate, and uninstall the service. Oracle provides a unique set of scripts for CCS, and packs them all in the archive.tgz file that you download from either Oracle Software Delivery Cloud or My Oracle Support. The download creates the following directory tree on the host.

Directory tree:

```
./ccs-<version>.<build>/install.pl
./ccs-<version>.<build>/ccs
./ccs-<version>.<build>/ccs/.build (hidden)
./ccs-<version>.<build>/ccs/.version (hidden)
./ccs-<version>.<build>/ccs/api
./ccs-<version>.<build>/ccs/api/KeyRsp.v1.json
./ccs-<version>.<build>/ccs/api/RegReq.v1.json
./ccs-<version>.<build>/ccs/api/RegRspv1.json
./ccs-<version>.<build>/ccs/api/TokenRsp.v1.json
./ccs-<version>.<build>/ccs/cfg
./ccs-<version>.<build>/ccs/cfg.v1.json
./ccs-<version>.<build>/ccs/img
./ccs-<version>.<build>/ccs/img/ccs-<version>.<build>.tar
./ccs-<version>.<build>/ccs/log
./ccs-<version>.<build>/ccs/perl
./ccs-<version>.<build>/ccs/perl/activate.pl
./ccs-<version>.<build>/ccs/perl/config.pl
./ccs-<version>.<build>/ccs/perl/deactivate.pl
./ccs-<version>.<build>/ccs/perl/uninstall.pl
./ccs-<version>.<build>/ccs/ssl
./ccs-<version>.<build>/ccs/ssl/ca
./ccs-<version>.<build>/ccs/ssl/ca/c_rehash
./ccs-<version>.<build>/ccs/ssl/ca/DigiCertGlobalRootCA.cer
./ccs-<version>.<build>/ccs/ssl/ca/DigiCertSHA256GlobalCaG2.cer
./ccs-<version>.<build>/ccs/ssl/ca/DigiCertSHA256GlobalRootG2.cer
./ccs-<version>.<build>/ccs/ssl/ca/DigiCertSHA2SecureServerCA.cer
```

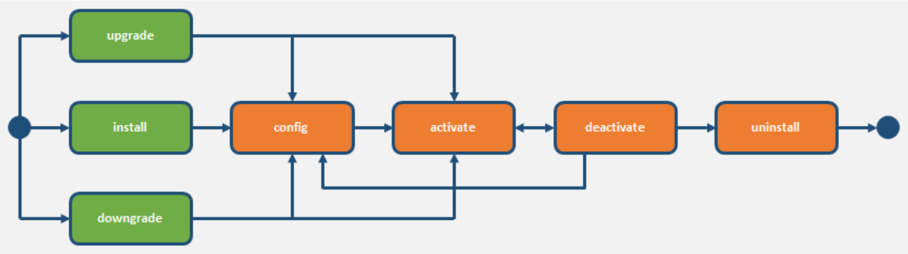
The initial installation process for the CCS includes running the scripts in the following order:

1. Install
2. Configure
3. Activate

After the initial installation you can use the various scripts to manage the CCS, as follows:

- Reconfigure and reactivate the installed version of the CCS, for example, if you want to change the host, configuration, or certificates.
- Deactivate and reactivate the existing configuration.
- Uninstall the CCS.
- Upgrade and downgrade the CCS version.

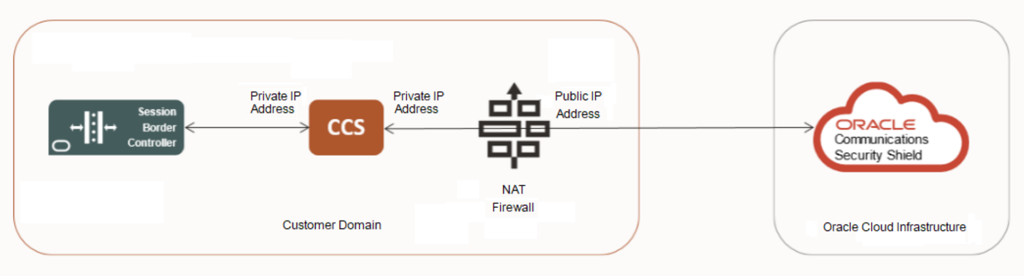
The following illustration shows the possible work flows for running the scripts:



For instructions for running the scripts, see the *Security Shield Installation and Maintenance Guide*.

CCS Configuration Behind NAT or a Firewall

Oracle recommends that you configure the Cloud Communication Service (CCS) to operate behind Network Address Translation (NAT) or a firewall.



Oracle designed the Oracle® Communications Security Shield Cloud Service (Security Shield) to contact the CCS using the value for the **"Server-FQDN"** configuration field in the CCS. The CCS supplies the **"Server-FQDN"** value when it registers with Security Shield. For example:

```
"Server-FQDN" : "ccs.useast.example.com"
```

You can set the **"Server-FQDN"** value as an FQDN or a static IP address that maps to the public interface of the NAT or firewall. Security Shield always uses port 443 for these connections, which requires any device placed between the CCS and Security Shield to dedicate port 443 to the CCS for all possible IP addresses resolved for the FQDN.

Security Shield Phone Number Format Requirements

Oracle® Communications Security Shield Cloud Service (Security Shield) requires the following conventions for phone numbers for inbound and outbound calls.

> **Note:**
>
> If your Session Border Controller does not use phone numbers in the E.164 format, Oracle may need to work with you before deploying Security Shield to determine how to normalize your phone numbers to work effectively with Security Shield.

- **Phone Number Format**
  The general number format convention is country code followed by the subscriber phone number <country code><subscriber phone number>. The subscriber phone number may include an area code and is typically seven to eleven digits long, depending on the national number conventions. Enter Phone numbers in the following ten-digit format where N is any digit from 2-9 (first digit of the area code and the local exchange) and X is any digit from 0-9: **NXX-NXX-XXXX** .

  > **Note:**
  >
  > The preceding example contains hyphens only to aid in understanding. Do not insert hyphens when entering the number in the "Number" field in the "Add Outbound Number" configuration.

- **Country Code**
  The country code can be up to three digits long. For international formatting, you may format the number with a + character (+<country code><subscriber phone number>, for example, +15551234567) or without the + character. For outbound calls to international destinations you can use either the + character or the international dialing prefix for your country. Check with your SIP trunk provider for the number format convention it supports. When formatting phone numbers for the Trusted Enterprise Calls subscription, which is valid only in North America, use one of the following methods to add the country code.
  - **Manual**—Add the +1 or 1 prefix to the number, for example +1NXXNXXXXXX or 1NXXNXXXXXX.
  - **Default to the United States and Canada**—Skip adding +1 or 1 to the number. Go to the Settings page and click Autonomous Threat Protection. Select "United States and Canada" for the "Service Domain Home Country." Security Shield will consider all phone numbers without a country code as "United States and Canada."
- You can use wild cards at the end of the phone number to indicate a range, except for Trusted Enterprise Calls. For example: To specify a seven digit phone number that begins with 91920, enter 91920xx.
- If you choose to configure the Presentation Number, you must use only the number format convention supported by the SIP trunk provider. When you use multiple SIP trunk providers, you must use a Presentation Number format that each SIP Trunk provider can support. For example, in the United States you use [country code][area code][local phone number] or the more commonly used [area code][local phone number]. In the European Union and United Kingdom you use [+][country code][area code][local phone number].

Number Cleansing

Use the following information to help you prepare phone numbers for Security Shield processing.

- Try to map to a country code to set the country.
- Remove any leading zeros from the phone number without the country code that may occur from configuration issues with trunk code or international dialing prefixes that are not removed.
- Map the next digits (prefix after the country code, after removing any leading zeros) to a carrier.
- Determine if the number length matches with the number plan (length for the prefix range.

When you receive calling number information (SIP INVITE, FROM, or PAI fields) containing a short phone number, incorrect format, or alphanumeric text such as "Restricted or "Anonymous", the reputation score may by negatively affected and can cause false positives for Reputation Score (High risk categories) and Threat Detection (Call Type). Oracle recommends that you use Security Shield Number Normalization, Non-Conforming E.641 Numbers guidelines, and Access Control Lists to avoid processing numbers with incorrect formats or alphanumeric text.

Session Router Support for Security Shield

The Oracle® Communications Security Shield Cloud Service (Security Shield) supports the Oracle Communications Session Router. The Session Router works in environments that use Oracle Session Border Controllers to integrate with Security Shield as well as environments that do not use Oracle Communications Session Border Controllers to integrate with Security Shield.

The Oracle Communications Session Router (OCSR) resides in the signaling core and directs traffic to and from other SIP signaling elements in the network, including Call Session Control Function servers.

Requirements

- Use S-Cz9.0.0 or higher.
- Install the Security Shield SPL on the external facing realm.
- Session Router runs in the session stateful configuration mode, only. Security Shield does not support other configurations and modes.

Deployment Considerations

Oracle supports the following models for deploying the Session Router.

- You can deploy the Session Router between the existing border element and SIP Trunk. You must enable the Security Shield SPL Plug-in on the external facing realm, which may require you to reconfigure the SIP Trunk end-point.
- You can deploy the Session Router inside the enterprise network. You must use two distinct realms on the Session Router if you send both inbound and outbound traffic from the same border element. You must configure Security Shield on the realm that will receive traffic from the SIP Trunk. The other realm must receive traffic from inside the Enterprise network. On the Session Router, traffic is routed between these two realms.
- See Oracle Communications Session Router for a product overview.
- See Session Border Controller Documentation for installation and configuration. See the *Platform Preparation and Installation Guide*, the *ACLI Configuration Guide*, and the *ACLI Reference Guide*.
- See Acme Packet Hardware Documentation for specifications and installation per platform.

Upgrade Information

The following topics provide important information you need to know the before an Oracle® Communications Security Shield Cloud Service (Security Shield) upgrade. For some releases, you might need to perform certain tasks before the upgrade.

New Version

For enabling your Session Border Controller or Session Router to pull policy updates from Security Shield for calls, you must upgrade to the relevant S-Cx release and patch levels. SCz9.1.0p6 (or higher, when on 9.1.x release) or SCz9.2.0p1 (or higher, when on 9.2.x release). See your Customer Support Representative for more information.

Older Versions

Older versions rely on Security Shield pushing updates to your session Border Controller or Session Router in separate API calls originated by Security Shield. The pulling capabilities allow you to simplify your network configuration, including firewalls and proxies, because you do not need to create pinholes that might allow unexpected API calls to reach your network.

Upgrade Information for the 23.1.0.0.0 Release

Oracle recommends that you review the following information about upgrades before using Oracle® Communications Security Shield Cloud Service (Security Shield).

Required Oracle® Communications Security Shield Cloud Service (Security Shield) Version

For the new policy update enhancement, you must upgrade to the relevant S-Cx release and patch levels. SCz9.1.0p6 (or higher, when on 9.1.x release) or SCz9.2.0p1 (or higher, when on 9.2.x release). See your Customer Support Representative for more information.

3 Security Shield Deployment Process and Procedures

Obtaining and installing the Oracle® Communications Security Shield Cloud Service (Security Shield) service requires a multi-step process that includes tasks for you to perform in the Oracle Cloud and on premises. New customers must take steps to establish and set up their Oracle Cloud account in addition to the procedures for installing the Security Shield service. See the following topics to guide you through the process.

Topics:

- Establish a Security Shield Subscription
- Download the Cloud Communication Service Software from MOS
- Configure TLS Certificates for the Cloud Communication Service
- Install, Configure, and Activate the Cloud Communication Service
- Configure TLS Certificates for the OCSBC
- Configure the Session Border Controller for the Security Shield Service

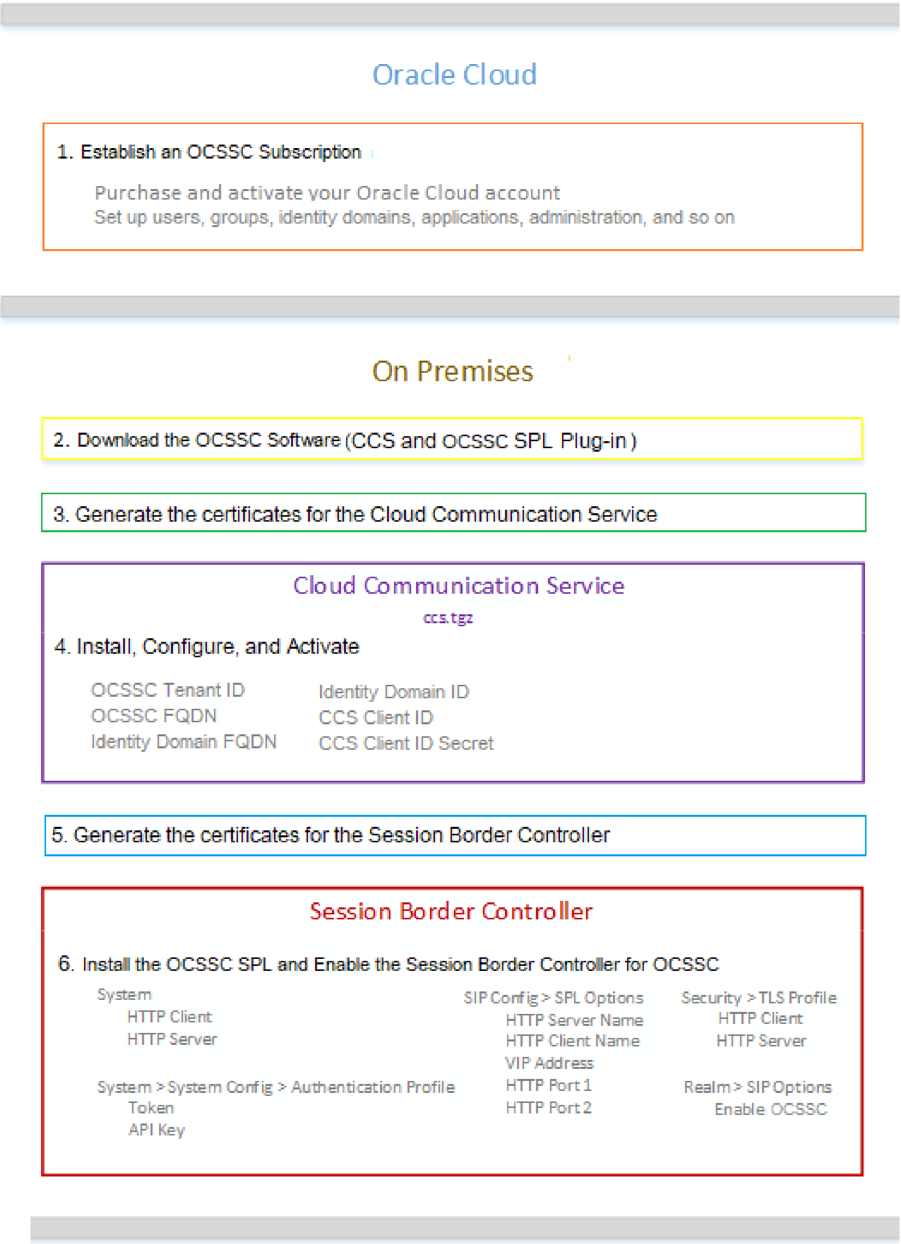
Security Shield Deployment Process

The high-level process for deploying the Oracle® Communications Security Shield Cloud Service (Security Shield) includes the following steps. You will perform some steps in the Oracle Cloud and others on-premises.

1. Oracle Cloud—Contact your Oracle Cloud sales representative to establish a subscription for Security Shield and activate your account. (Security Shield is not available for purchase online in the Oracle Marketplace.) See Establish a Security Shield Subscription.
2. On premises—Log on to Oracle Service Delivery Cloud at edelivery.oracle.com using the Customer Support Identifier number that you received in your Oracle Cloud Welcome email and download the Security Shield software. (Cloud Communication Service and Security Shield SPL file) See Download the Cloud Communication Service Software from MOS.
3. On-premises—Generate the TLS certificates for the Cloud Communication Service. The installation script requires a pem file and a key file for both the LAN side and the WAN side. Configure TLS Certificates for the Cloud Communication Service.
4. On premises—Install the Cloud Communication Service with the install, configure, and activate scripts provided in the software download. See either Install, Configure, and Activate the Cloud Communication Service.
5. On-premises—Generate the TLS certificate for the Session Border Controller and install it on the SBC. Configure TLS Certificates for the OCSBC.
6. On premises—Install the Security Shield SPL on the Session Border Controller (SBC) and enable Security Shield on the SBC, which registers the SBC with the Cloud

Communication Service. See Configure the Session Border Controller for the Security Shield Service .

The following diagram illustrates the deployment process and shows the parameters you need to set in each Security Shield component to establish the service.



Oracle Cloud

1. Establish an OCSSC Subscription

Purchase and activate your Oracle Cloud account
Set up users, groups, identity domains, applications, administration, and so on

On Premises

2. Download the OCSSC Software (CCS and OCSSC SPL Plug-in)

3. Generate the certificates for the Cloud Communication Service

Cloud Communication Service
ccs.tgz

4. Install, Configure, and Activate

OCSSC Tenant ID	Identity Domain ID
OCSSC FQDN	CCS Client ID
Identity Domain FQDN	CCS Client ID Secret

5. Generate the certificates for the Session Border Controller

Session Border Controller

6. Install the OCSSC SPL and Enable the Session Border Controller for OCSSC

System
HTTP Client
HTTP Server

System > System Config > Authentication Profile
Token
API Key

SIP Config > SPL Options
HTTP Server Name
HTTP Client Name
VIP Address
HTTP Port 1
HTTP Port 2

Security > TLS Profile
HTTP Client
HTTP Server

Realm > SIP Options
Enable OCSSC

Next Steps

- Oracle Cloud—Log on to theSecurity Shield Dashboard and customize the threat protection settings. See "Customize the Security Shield Autonomous Threat Protection Settings" in the *Security Shield User's Guide*.
- Oracle Cloud—Log on to the Security Shield Dashboard and create managed phone number lists. See the "Add Phone Numbers to Policy Rules" in the *Security Shield User's Guide*.

Establish a Security Shield Subscription

To obtain the Oracle® Communications Security Shield Cloud Service (Security Shield), contact Oracle Cloud Sales to purchase a Cloud Services Agreement and the Security Shield service subscription. Oracle does not offer the Security Shield service as an online purchase in the Oracle Cloud Marketplace. You must purchase through Oracle Sales.

Establishing an Security Shield service subscription is a multi-step process. Use the information provided in the following links to guide you through the process.

1. Go to Oracle Communications Security Shield Cloud and click **Contact Sales** for information about how to purchase a subscription to Security Shield.
2. Go to Activate Your Cloud Account for instructions to activate your subscription.
3. Go to Oracle Cloud Infrastructure Identity and Access Management for information about how to manage your Users and Groups, Identity Domains, Applications, Administration, and more.

Next Steps

- Download the Security Shield Software

Download the Security Shield Software

Before you can install the Cloud Communications Service (CCS) and the OCSSC SPL plug-in file on-premises, you must download the software on to the host. Only authorized customers with a valid password may download the software. Note that the download includes both the CCS software and the Oracle® Communications Security Shield Cloud Service (Security Shield) SPL plug-in file that you must install on the Session Border Controller.

The following procedure explains how to download the CCS software and SPL plug-in file from Oracle Software Delivery Cloud.

> **Note:**
>
> If you prefer, you can download the CCS software and SPL plug-in file from My Oracle Support (MOS). See Download the Cloud Communication Service Software from MOS.

Procedure

1. Log on to https://edelivery.oracle.com with the Customer Support Identifier number that you received in your "Welcome" email from Oracle.
2. Search for Cloud Communications Service software.
3. Add the Cloud Communications Service software to your shopping cart.
4. Check out and download the Cloud Communications Service software file to your host.

   Oracle creates the /opt/oracle directory and necessary sub-directories on the host for the CCS software.

Next Steps

- Install, Configure, and Activate the Cloud Communication Service

Configure TLS Certificates for the Cloud Communication Service

The Cloud Communication Service (CCS) uses TLS to encrypt and secure your data on the Local Area Network between CCS and the Session Border Controller (SBC) and on the Wide Area Network (WAN) between CCS and Oracle® Communications Security Shield Cloud Service (Security Shield). The WAN and LAN connections both require a pem format certficate file and its matching key file. You also need the intermediateCA certificate file when you run the activate script in the "Install, Configure, and Activate the Cloud Communication Service" procedure.

Use the following procedure to generate the required pem and key TLS certificates for your LAN and WAN.

1. Create a san.cnf file with your local information and set the IP.1 parameter to the IP address you want the CCS to use on the LAN side.

   ```
   {req}
   default_bits = 2048
   distinguished_name = req_distinguished_name
   req-extensions = req_ext
   [req_distinguished_name]
   countryName = Country Name (2 letter code)
   stateOrProvinceName = State or Province Name (full name)
   localityName = Locality Name (For example, city)
   organizationName = Organization Name (For example,company)
   commonName = Common Name (For example, server FQDN or YOUR name)
   {req_ext]
   subjectAltName = @alt_names
   [alt_names]
   IP.1 = <CCS LAN-side IP Address>
   ```

2. Use OpenSSL to generate the Certificate Signing Request and key file using the san.cnf file you created above.

   ```
   openssl req -out <path where to create sslcert.csr> -newkey
   rsa:2048 -nodes -keyout
    <path where to create lan-key.pem> -config <path to san.cnf>
   ```

3. Use OpenSSL to generate a signed certificate pem file for the associated Certificate Signing Request, setting the expiration period you want with the -days parameter (the following CA is for the LAN-side signing CA).

   ```
   openssl ca -config <path to openssl.cnf> -extensions server_cert -
   days <set certificate
         expiration in days> -notext -md sha256 -in <path to
   sslcert.csr>  -out <path where to create
         lan-cert.pem>
   ```

4. Verify the certificate information and that the certificate states the correct IP address in the IP.1 field.

   ```
   openssl x509 -noout -text -in <path to lan-cert.pem>
   ```

5. Modify the san.cnf file and change the IP.1 parameter to the IP address you want the CCS to use for the WAN side. The common name is the FQDN to use for the WAN-side.

```
[ req ]
default_bits = 2048
distinguished_name = req_distinguished_name
req_extensions     = req_ext
[ req_distinguished_name ]
countryName = Country Name (2 letter code)
stateOrProvinceName = State or Province Name (full name)
localityName = Locality Name (For example, city)
organizationName = Organization Name (For example, company)
commonName = Common Name (For example, server FQDN or YOUR
name)
[ req_ext ]
subjectAltName = @alt_names
[alt_names]
IP.1 = CCS WAN SIDE IP ADDRESS
```

6. Use OpenSSL to generate the Certificate Signing Request and key file using the san.cnf file created above.

```
openssl req -out <path where to create sslcert.csr> -newkey rsa:2048 -
nodes -keyout <path where
      to create wan-key.pem> -config <path to san.cnf>
```

7. Send the sslcert.csr to a trusted CA for signing. Oracle assumes that the returned certificate is named wan-cert.pem.

8. Verify the certificate information and that the certificate states the correct IP address in the IP.1 field.

```
 openssl x509 -noout -text -in <path to wan-cert.pem>
```

Next steps

- Acquire the intermediateCA certificate. (LAN-side signing CA "lan-ca-cert.pem")
- Run the CCS activation script which requires the LAN and WAN certificates and matching keys, plus the LAN-side signing CA certfiicate. Install, Configure, and Activate the Cloud Communication Service.

Install, Configure, and Activate the Cloud Communication Service

The Cloud Communication Service (CCS) installation procedure requires the archive file containing the installation, configuration, and activation scripts that you downloaded from Oracle onto your host hardware.

Oracle recommends running the three scripts consecutively in one session the first time you install CCS. For that reason, this procedure includes the prerequisites and steps for running the scripts sequentially. Oracle recommends that you install the CCS behind Network Address Translation (NAT) or a firewall. See CCS Configuration Behind NAT or a Firewall.

CCS Installation Prerequisites
Do the following before performing the CCS installation procedure.

System Prerequisites

- Ensure that the host meets Operation System and resource requirements. Version Requirements for External Components.
- Install Perl v5.16.3 or higher on the host.
- Ensure that you have Root access or equivalent Super User privileges.
- Ensure that the Docker (v18.09.1 or higher) or Podman (v4.4.1 or higher) package is installed and that the daemon is running as a service if you use Docker.
- Ensure that you synchronize the CCS Docker host and the Session Border Controller with the Network Time Protocol (NTP) server, if you use Docker.

Installation Script Prerequisites

- Ensure that there is no CCS installation existing on the hardware. See the last step in this procedure for instructions.
- Download the archive file (ccs-<version>.tgz) from Oracle Service Delivery Cloud at edelivery.oracle.com, which includes all of the scripts, to the host server.

Configuration Script Prerequisites

- Generate the local API key and local API alternate key. See the *Security Shield Security and Privacy Guide* for instructions for generating keys.
- Host WAN FQDN provided by the customer
- Host WAN IP Address provided by the customer
- Host LAN IP Address provided by the customer
- Identity Domain FQDN provided by Security Shield through the Cloud Communications Service Configuration link on the Settings tab
- Identity Domain ID provided by Security Shield through the Cloud Communications Service Configuration link on the Settings tab
- Security Shield FQDN provided by Security Shield through the Cloud Communications Service Configuration link on the Settings tab
- Security Shield Tenant ID provided by Security Shield through the Cloud Communications Service Configuration link on the Settings tab
- Security Shield API Key provided by the customer (the CCS API key)
- Security Shield API Key Alternate provided by the customer
- CCS Client ID provided by Security Shield through the Cloud Communications Service Configuration link on the Settings tab
- CCS Client Secret provided by Security Shield through the Cloud Communications Service Configuration link on the Settings tab

Activation Script Prerequisites

- CCS is installed and not activated
- CCS JSON configuration (cfg.json) provided by customer from prior configuration step
- LAN/OAM server certificate provided by customer
- LAN/OAM server private key provided by customer
- LAN/OAM server signing certificate provided by customer

- (Optional)—WAN server certificate provided by customer. Use when you want ground-to-cloud communication, which allows mid-call updates. Omit when you do not want cloud-to-ground communication, which does not allow mid-call updates.
- (Optional)—WAN server private key provided by customer. Use when you want ground-to-cloud communication,which allows mid-call updates. Omit when you do not want cloud-to-ground communication, which does not allow mid-call updates.

> **Note:**
>
> See the *Security Shield Security and Privacy Guide* for information about certificate management and deploying CCS behind Network Address Translation (NAT) or a firewall.

Procedure

The CCS installation script sets the defaults for the LAN, WAN, and OAM server instances to the address of 0.0.0.0 and the ports to 8000, 443, and 2000, respectively. After you execute config.pl, CCS creates the cfg.json file and sets the WAN listening port to the default 443. If you provide the WAN certificate and private key in the activate.pl script configuration, the WAN listening port remains 443. If you do not provide the WAN certificate and private key, CCS changes the port to 9000.

If you modify the WAN port in cfg.json to any value other than 443 or 9000, CCS retains that value even when you provide the WAN certificate and private key. Ensure that if you change the default ports, they do not conflict with the Oracle Management Cloud Engine (OMCE) ports.

> **Note:**
>
> If you change the Cloud Communication Service (CCS) public IP address (WAN interface), it may take up to twenty four hours for mid-call updates to resume.

In the following procedure, wait for each script to finish running successfully before running the next one.

> **Note:**
>
> The Cloud Communication Service (CCS) does not support simultaneous use of the same CCS instance by different services, for example Security Shield and Oracle Session Delivery Manager Cloud (OSDMC). You must configure each CCS instance to support only one service.

1. Log on to the server at root.
2. Unpack the ccs-<version>.<build>.tgz archive.

```
tar -xvzf ccs-<version>.<build>.tgz
```

The system creates the ccs-<version> directory and copies the unpacked files there in the following directory tree.

- /opt/oracle
- /opt/oracle/ccs
- /opt/oracle/ccs/.build (hidden)
- /opt/oracle/ccs/.version (hidden)
- /opt/oracle/ccs/api
- /opt/oracle/ccs/api/KeyRsp.v1.json
- /opt/oracle/ccs/api/RegReq.v1.json
- /opt/oracle/ccs/api/RegRspv1.json
- /opt/oracle/ccs/api/TokenRsp.v1.json
- /opt/oracle/ccs/cfg
- /opt/oracle/ccs/cfg.v1.json
- /opt/oracle/ccs/img
- /opt/oracle/ccs/img/ccs-<version>.<build>.tar
- /opt/oracle/ccs/log
- /opt/oracle/ccs/perl
- /opt/oracle/ccs/perl/activate.pl
- /opt/oracle/ccs/perl/config.pl
- /opt/oracle/ccs/perl/deactivate.pl
- /opt/oracle/ccs/perl/uninstall.pl
- /opt/oracle/ccs/ssl
- /opt/oracle/ccs/ssl/ca
- /opt/oracle/ccs/ssl/ca/c_rehash
- /opt/oracle/ccs/ssl/ca/DigiCertGlobalRootCA.cer
- /opt/oracle/ccs/ssl/ca/DigiCertSHA256GlobalCaG2.cer
- /opt/oracle/ccs/ssl/ca/DigiCertSHA256GlobalRootG2.cer
- /opt/oracle/ccs/ssl/ca/DigiCertSHA2SecureServerCA.cer

3. At the prompt, do the following:

   a. Type **cd ccs-<version>**, and press Enter.

   b. Type **ls**

   c. Type **./ccs-<version>.<build> /install.pl**

   ```
   # cd ccs-<version>
   # ls
   # ccs install.pl upgrade.pl
   ```

4. At the prompt, type **./install.pl**, and press Enter.

The system checks for an existing installation and, if none exits, asks if you want to proceed with the installation.

```
# ./install.pl
------------------------------------------------------
Oracle Cloud Communications Service, (c) 2020 Oracle
CCS <version> activate.pl @ <date> <time>
------------------------------------------------------
Checking pre-conditions...
OK.
Use Docker or Podman (d/p) :
Proceed with install (y/n) :
```

5. Type **d for Docker or p for Podman** and type **y**.

   The system installs CCS in the `/opt/oracle/` directory and displays a success message upon completion.

```
Installing...
Success.
```

6. At the prompt, type **/opt/oracle/ccs/perl/config.pl**, press Enter, and specify each of the following attributes.

```
# /opt/oracle/ccs/perl/config.pl
--------------------------------------------------------------------------
-----
Oracle Cloud Communications Service, (c) 2020 Oracle
CCS <version> <build> config.pl @ <date> <time>
--------------------------------------------------------------------------
-----
Please specify each attribute...
Host WAN FQDN             : host-wan-fqdn
Host WAN IP address       : host-wan-IP
Host LAN IP address       : host-lan-ip
Identity Domain FQDN      : idcs-fqdn
Identity Domain ID        : idcs-tenant-id
OCSSC subscriber (y/n)    : y
Security Shield FQDN      : ocssc-fqdn
Security Shield Tenant ID : occsc-tenantid
Security Shield API Key   : ocssc-apikey
Security Shield API Key Alternate    : ocssc-apialternate-key
Security Shield CCS Client ID        : ocssc-idcs-clientid
Security Shield CCS Client Secret    : ocssc-idcs-clientsecret
OSDMC subscriber (y/n)    : n
Proceed with config (y/n) : y
Generating local cfg.json...
Success.
# ls cfg.json
cfg.json
```

Host WAN FQDN	Set the host FQDN from your WAN-FQDN and Registration Client-Device-Name.

Host WAN IP address	Set the host WAN IP address. (Provided by the customer.)
Host LAN IP address	Set the host LAN IP address. (Provided by the customer.)
Identity Domain FQDN	Set the Identity Domain FQDN (Provided by Security Shield through the Cloud Communications Service Configuration link on the Settings tab.)
Identity Domain ID	Set the Identity Domain ID (Provided by Security Shield through the Cloud Communications Service Configuration link on the Settings tab.)
Security Shield subscriber (y/n)	Type y.
Security Shield FQDN	Set the OCSSC FQDN (Provided by Security Shield through the Cloud Communications Service Configuration link on the Settings tab.)
Security Shield Tenant ID	Set the OCSSC tenant ID (Provided by Security Shield through the Cloud Communications Service Configuration link on the Settings tab.)
Security Shield API Key	Set the local OCSSC API key that you generated. Valid syntax: alphanumeric. Oracle recommends using the UUID.
Security Shield API Key Alternate	Set the local OCSSC API key that you generated.
Security Shield CCS Client ID	Set the CCS Client ID (Provided by Security Shield through the Cloud Communications Service Configuration link on the Settings tab.)
Security Shield CCS Client ID Secret	Set the CCS client ID secret. (Provided by Security Shield through the Cloud Communications Service Configuration link on the Settings tab.)
OSDMC subscriber (y/n)	Type n.

After you enter "n" for OSDMC subscriber, the system asks if you want to proceed with the configuration.

```
Proceed with config (y/n) :
```

7. Type **y**, and press Enter.

   The system generates the cfg.json file.

   ```
   Generating local cfg.json...
   Success.
   ```

8. At the prompt, type **/opt/oracle/ccs/perl/activate.pl**, and press Enter.

   The system verifies that an installed CCS exists.

   ```
   # /opt/oracle/ccs/perl/activate.pl
   --------------------------------------------------------------------
   -----------
   ```

```
Oracle Cloud Communications Service, (c) 2020 Oracle
CCS <version> <build> activate.pl @ <date> <time>
--------------------------------------------------------------------------
-----
Checking pre-conditions...
Ok.
Please specify import path for each file...
CCS JSON configuration              : ./cfg.json
LAN /OAM server certificate         : ./lan-cert.pem
LAN /OAM private key                : ./lan-key.pem
LAN /OAM server signing certificate : ./lan-ca-cert.pem
WAN server certificate              : ./wan-cert.pem
WAN server private key              : ./wan-key.pem
Proceed with activate (y/n) : y
Importing...
Activating...
Success.
```

9. Specify the relative or absolute path to the locations for the following files. The following example uses a relative path to the required files. You can use a file name of your own choosing,but the certificate file names must match those from the Configure TLS Certificates for the Cloud Communication Service procedure. The path is relative to wherever you choose to execute the activate.pl script.

   > **Note:**
   >
   > See the *Security Shield Security and Privacy Guide* for information about certificates and keys.

CCS JSON configuration	./cfg/cfg.json (This is the file you generated in step 6 and 7.)
LAN /OAM server certificate	./ssl/lan-cert.pem. (Provided by the customer.)
LAN /OAM server private key	./ssl/lan-key.pem. (Provided by the customer.)
LAN /OAM server signing certificate	./ssl/lan-ca-cert.pem. (Provided by the customer.)
(Optional) WAN server certificate	• To enable mid-call updates: ./ssl/wan-cert.pem. (Provided by the customer.) • To disable mid-call updates: Leave empty.
(Optional) WAN server private key	• To enable mid-call updates: ./ssl/wan-key.pem. (Provided by the customer.) • To disable mid-call updates: Leave empty.

   The system asks if you want to proceed with the activation.

10. Type: **y**.

The system displays the status of the activation.

```
Proceed with activate? y/n : y
Importing...
Activating...
Success,  ccs-<build>.<version> is up and running.

CONTAINER ID
IMAGE
COMMAND              CREATED                 STATUS
PORTS            NAMES
5ab14bc101bf  example.com:/apps/cgbu/cocssc/ccs-core:ccs-
<build>.<version>  --cfg=/mnt/cfg/cf...  Less than a second ago  Up
Less than a second              <ccs-build>.<version>
```

11. (Optional) Use Docker or Podman to check your work.

    a. Run the appropriate command, as shown below in the following examples.

```
# docker image ls
REPOSITORY
TAG                IMAGE ID       CREATED     SIZE
example.com:/apps/cgbu/ocssc/ccs-core    ccs-<version> <build>
74e9303190d3   12 hours ago  132MB


# podman image ls
REPOSITORY                                       TAG
IMAGE ID      CREATED      SIZE
example.com:/apps/cgbu/ocssc/ccs-core  ccs-<build>.<version>
72c445e6016a  22 hours ago  158 MB
```

    b. At the prompt type: docker ps or podman ps, and press Enter to list the running images. The following code blocks show examples.

```
# docker ps
CONTAINER ID       IMAGE            COMMAND
CREATED       STATUS         PORTS             NAMES
0fa66a7dc1bb    74e9303190d3  "ccs.exe --config /m..."   6 hours
ago   Up 3 hours                ccs-<build>.<version>


# podman ps
CONTAINER ID
IMAGE                                      COMMAND
CREATED       STATUS        PORTS             NAMES
fc59a256efb0  example.com:/apps/cgbu/ocssc/ccs-core :ccs-
<build>.<version>  --cfg=/mnt/cfg/cf...  4 minutes ago  Up 4
minutes              ccs-<build>.<version>
```

Configure TLS Certificates for the OCSBC

The process for configuring a certificate on the Oracle Communications Session Border Controller (OCSBC) requires the following steps.

1. Configure a certificate record on the SBC. See Configure a Certificate Record.
2. Generate a certificate request by the SBC. See Generate a Certificate Request.
3. Import the certificate into the SBC. See Import a Certificate Using SFTP or Import a Certificate Using the ACLI.
4. Reboot the system.

Configure a Certificate Record

Use the certificate-record object to add a certificate record to the Oracle® Communications Security Shield Cloud Service (Security Shield). The certificate record configuration represents either the end-entity or the Certificate Authority (CA) certificate on the Security Shield.

When you configure a certificate for the E-SBC, the name that you enter must be the same as the name that you use when you generate a certificate request. If configuring for an end stations CA certificate for mutual authentication, the certificate name must be the same name used during the import procedure.

- If this certificate record is used to present an end-entity certificate, associate a private key with this certificate record by using a certificate request.
- If this certificate record is created to hold a CA certificate or certificate in PKCS12 format, a private key is not required.

1. Access the **certificate-record** configuration element.

```
ORACLE# configure terminal
ORACLE(configure)# security
ORACLE(security)# certificate-record
ORACLE(certificate-record)#
```

2. Do the following:

   name—Enter the name of the certificate record. Required.

   country—Enter the name of the country. Default: U.S.

   state—Enter the name of the state of for the country. Default: MA.

   locality—Enter the name of the locality for the state. Default: Burlington.

   organization—Enter the name of the organization holding the certificate. Default: Engineering.

   unit—Enter the name of the unit for the holding the certificate within the organization.

   common-name—Enter the common name for the certificate record.

   key-size—Enter the size of the key for the certificate. Default:1024 Valid values: 512 | 2048 | 4096.

   alternate-name—Enter the alternate name of the certificate holder.

   key-usage-list—Enter the usage extensions you want to use with this certificate record. This parameter can be configured with multiple values, and it defaults to the combination of digitalSignature and keyEncipherment. For a list of possible values and their descriptions, see "Key Usage Control."

extended-key-usage-list—Enter the extended key usage extensions you want to use with this certificate record. Default: serverAuth. For a list of possible values and their descriptions, see "Key Usage Control."

3. Type **done** to save your configuration.

To verify a certificate record, see "Security" in the *ACLI Configuration Guide*.

Generate a Certificate Request

Using the ACLI **generate-certificate-request <record-name>** command allows you to generate a private key and a certificate request in PKCS10 PEM format.

> **Note:**
>
> You can only perform this task after you configure a certificate record.

The Oracle® Communications Security Shield Cloud Service (Security Shield) stores the private key that is generated in the certificate record configuration in 3DES encrypted form with an internally generated password. The Security Shield displays the PKCS10 request in PEM (Base64) form.

You use this command for certificate record configurations that hold end-entity certificates. If you have configured the certificate record to hold a CA certificate, then you do not need to generate a certificate request because the CA publishes its certificate in the public domain. You import a CA certificate by using the ACLI **import-certficate <certficate-record-name>** command.

The **generate-certificate-request** command sends information to the CA to generate the certificate, but you cannot have Internet connectivity from the Security Shield to the Internet. You can access the Internet through a browser such as Internet Explorer if it is available, or you can save the certificate request to a disk and then submit it to the CA.

To run the applicable command, you must use the value you entered in the name parameter of the certificate record configuration. You run the command from the main Superuser mode command line, and then save and activate the configuration.

```
ACMEPACKET# security certificate request acmepacket
Generating Certificate Signing Request. This can take several
minutes....

-----BEGIN CERTIFICATE REQUEST-----

MIIB2jCCAUMCAQAwYTELMAkGA1UEBhMCdXMxCzAJBgNVBAgTAk1BMRMwEQYDVQQH
EwpCdXJsaW5ndG9uMRQwEgYDVQQKEwtFbmdpbmVlcmluZzEMMAoGA1UECxMDYWJj
MQwwCgYDVQQDEwNhYmMwgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBALOMLHo8
/qIOddIDVuqot0Y72l/BfH8lolRKmhZQ4e7sS+zZHzbG8phzmzhfOSECnZiA2bEo
f+Nti7e7Uof4lLwiYl9fvhURfzhENOKThAPKPiJCzBBglTITHTYal00Cq2fj5A8B
ZcuAHj7Vp5wP2zpz6EUTFpqTDMLVdwJGJrElAgMBAAGgOTAMBgNVHRExBRMDZGVm
MCkGA1UdDzEiEyBkaWdpdGFsU2lnbmF0dXJlLGtleUVuY2lwaGVybWVudDANBgkq
hkiG9w0BAQUFAAOBgQAtel4ZSLI8gqgMzodbYwgUHUGqTGeDzQDhJV5fKUXWeMFz
JsTmWn5Gy/kR4+Nq274G14fnk00fTAfMtgQ5aL3gM43TqaPOTZjJ6qgwuRKhoBPI
7hkovkgAxHge7wClghiAp/ELdl7tQ515k04BMd5f/fxG7nNiu8iEg7PO0OIBgg==
-----END CERTIFICATE REQUEST-----
```

```
WARNING: Configuration changed, run "save-config" command.
ACMEPACKET# save config
copying file /code/config/dataDoc.gz -> /code/config/dataDoc_3.gz
copying file /code/config/tmp/editing/dataDoc.gz ->
/code/config/dataDoc.gz
Save complete
ACMEPACKET# activate config
activate complete
```

Import a Certificate Using the ACLI

For an end-entity certificate, after a certificate is generated using the ACLI security certificate request command, submit the request to a CA for generation of a certificate in PKCS7 or X509v3 format. When the certificate has been generated, you can import it into the Oracle® Communications Security Shield Cloud Service (Security Shield) using the security certificate import command.

The syntax is:

```
ACMEPACKET # security certificate import [try-all | pkcs7 | pkcs12 |
x509] [certificate-record file-name]
```

To import a certificate:

1. When you use the **import-certificate <certificate-record-name>** command, you can specify whether you want to use PKCS7, PKCS12, X509v3 format, or try all. In the command line, you enter the command, the format specification, and the name of the certificate record. The Security Shield prompts you to enter the certificate in PEM format. Paste the certificate in the ACLI. For example:

   ```
   ACMEPACKET# security certificate import try-all acmepacket
   The following displays:
   Please enter the certificate in the PEM format.
   Terminate the certificate with ";" to exit.......
   -----BEGIN CERTIFICATE----
   VMIIDHzCCAoigAwIBAgIIAhMCUACEAHEwDQYJKoZIhvcNAQEFBQAwcDELMAkGA1UE
   BhMCVVMxEzARBgNVBAgTCkNhbGlmb3JuaWExETAPBgNVBAcTCFNhbiBKb3NlMQ4w
   DAYDVQQKEwVzaXBpdDEpMCcGA1UECxMgU2lwaXQgVGVzdCBDZXJ0aWZpY2F0ZSBB
   dXRob3JpdHkwHhcNMDUwNDEzMjEzNzQzWhcNMDgwNDEyMjEzNzQzWjBUMQswCQYD
   VQQGEwJVUzELMAkGA1UECBMCTUExEzARBgNVBAcTCkJ1cmxpbmd0b24xFDASBgNV
   BAoTC0VuZ2luZWVyaW5nMQ0wCwYDVQQDEwRhY21lMIGfMA0GCSqGSIb3DQEBAQUA
   A4GNADCBiQKBgQCXjIeOyFKAUB3rKkKK/+59LT+rlGuW7Lgc1V6+hfTSr0co+ZsQ
   bHFUWAA15qXUUBTLJG13QN5VfG96f7gGAbWayfOS9Uymold3JPCUDoGgb2E7m8iu
   vtq7gwjSeKNXAw/y7yWy/c04FmUD2U0pZX0CNIR3Mns5OAxQmq0bNYDhawIDAQAB
   o4HdMIHaMBEGA1UdEQQKMAiCBnBrdW1hcjAJBgNVHRMEAjAAMB0GA1UdDgQWBBTG
   tpodxa6Kmmn04L3Kg62t8BZJHTCBmgYDVR0jBIGSMIGPgBRrRhcU6pR2JYBUbhNU
   2qHjVBShtqF0pHIwcDELMAkGA1UEBhMCVVMxEzARBgNVBAgTCkNhbGlmb3JuaWEx
   ETAPBgNVBAcTCFNhbiBKb3NlMQ4wDAYDVQQKEwVzaXBpdDEpMCcGA1UECxMgU2lw
   aXQgVGVzdCBDZXJ0aWZpY2F0ZSBBdXRob3JpdHmCAQAwDQYJKoZIhvcNAQEFBQAD
   gYEAbEs8nUCi+cA2hC/lM49Sitvh8QmpL81KONApsoC4Em24L+DZwz3uInoWjbjJ
   QhefcUfteNYkbuMH7LAK0hnDPvW+St4rQGVK6LJhZj7/yeLXmYWIPUY3Ux4OGVrd
   2UgV/B2SOqH9Nf+FQ+mNZOlL7EuF4IxSz9/69LuYlXqKsG4=
   -----END CERTIFICATE-----;
   ```

```
Certificate imported successfully....
WARNING: Configuration changed, run "save-config" command.
```

2. Enter **save-config** to save the configuration.

   ```
   ACMEPACKET# save-config
   copying file /code/config/dataDoc.gz -> /code/config/dataDoc_3.gz
   copying file /code/config/tmp/editing/dataDoc.gz ->
   /code/config/dataDoc.gz
   Save complete
   ```

3. Enter **activate-config** to activate as the current configuration.

   ```
   ACMEPACKET# activate-config
   activate complete
   ```

> **Note:**
>
> For importing a certificate using SFTP, see the Security section of the *ACLI Configuration Guide* for your Security Shield model.

Import a Certificate Using SFTP

You can put the certificate file in the directory /ramdrv and execute the **import-certificate** command, or you can paste the certificate in PEM/Base64 format into the ACLI. If you paste the certificate, you may have to copy and paste it a portion at a time, rather than pasting the whole certificate at once.

1. SFTP the certificate file to the Oracle® Communications Security Shield Cloud Service (Security Shield) (directory /ramdrv). For the following example, suppose the name of the certificate file is cert.pem.
2. When the certificate is successfully transferred to the Security Shield, run the **import-certificate** command.

   The syntax is:

   ```
   ACMEPACKET# import-certificate [try-all|pkcs7|x509] [certificate-
   record file-name]
   ```

   Example results:

   ```
   ACMEPACKET# import-certificate try-all acme cert.pem
   Certificate imported successfully....
   WARNING: Configuration changed, run "save-config" command.
   ```

3. Save the configuration.

   ```
   ACMEPACKET# save-config
   Save-Config received, processing.
   waiting 1200 for request to finish
   Request to 'SAVE-CONFIG' has Finished,
   ```

```
Save complete
Currently active and saved configurations do not match!
To sync & activate, run 'activate-config' or 'reboot activate'.
```

4. Synchronize and activate the configurations.

```
ACMEPACKET# activate-config
Activate-Config received, processing.
waiting 120000 for request to finish
Add LI Flows
LiSysClientMgr::handleNotifyReq
H323 Active Stack Cnt:  0
Request to 'ACTIVATE-CONFIG' has Finished,
Activate Complete
ACMEPACKET#
```

Configure the Session Border Controller for the Security Shield Service

Specify the following information on the Session Border Controller (SBC) to complete the Oracle® Communications Security Shield Cloud Service (Security Shield) installation by setting security parameters and SPL options for communications between the SBC and Oracle® Communications Security Shield Cloud Service components.

Prerequisites

- Create the TLS profile that you want to use for the Security Shield HTTP client and server. See "Configure a TLS Profile" in the *ACLI Configuration Guide*.
- Create the authentication profile that you want to use for the Security Shield HTTP client and server. See the "Security" chapter in the *ACLI Configuration Guide*.
- Install the Security Shield SPL plug-in, which is included in the CCS download package. See Add the Security Shield SPL Plug-in.

Procedure

1. Go to **Configuration** on the SBC.
2. Go to **System**, **HTTP Client**, and set the following parameters.

Name	Set the name of the HTTP client.
State	Set to enable. Default: Enable.
Realm	Set the name of the realm to send requests on.
IP Address	Set the IP address of the HTTP client.
TLS Profile	Set the TLS profile that want this HTTP client to use.
Auth Profile	Set the name of the authentication profile that you want the HTTP client to use.

3. Go to **System**, **HTTP Server**, and set the following parameters.

Name	Set the name of the HTTP server.
State	Set to enable. Default: Enable.

Realm	Set the name of the realm to send requests on.
IP Address	Set the IP address of the HTTP server.
Inactivity Timeout	Specify the amount of time for the system to wait before timing out for inactivity. Default: 5 minutes. Valid values: 0-20 minutes.
HTTP State	Set to enable HTTP.
HTTP Port	Set the number of the HTTP port. Default: 80. Valid values: 1-65535. Caution: Do not use port 80 for Cz8.3.0.x Enterprise releases. Port 80 conflicts with the appweb server settings, resulting in unexpected behavior. Specify a different HTTP port here.
HTTPS State	Set to enable HTTPS.
HTTPS Port	Set the number of the HTTPS port. Default: 443. Valid values: 1-65535. Caution: Do not use port 443 for Cz8.3.0.x Enterprise releases. Port 443 conflicts with the appweb server settings, resulting in unexpected behavior. Specify a different HTTPS port here. This restriction does not apply to any Enterprise releases newer than 8.3.x, for example 8.4.x and 9.x.x.
TLS Profile	Set the TLS profile that you want this HTTP server to use.
Auth Profile	Set the name of the authentication profile that you want the HTTP server to use.

4. Go to **Security**, **Authentication Profile**, and set the following parameters.

Name	Set the name for this profile.
Authentication Scheme	Set the authentication scheme. Default: Bearer.
Pre-shared Key	Set the encrypted password for this profile.

5. Go to **Media Manager**, **Realm Config**, **SPL Options**, and add `ocssEnabled` to enable the Security Shield service in the security lookups for each realm that you want to use the Security Shield service.

6. Under **Session Router**, **SIP Config**, **SPL Options** enter the ocss-service-address for each CCS instance (up to three) in curly brackets. Note : {ipAddr1:port1; httpClientConfigName} can repeat up to a total of three times for three different CCSes with different IP:port options and optionally different httpClient Names (you can use the same client for all three CCSes).

   The SPL Option

```
ocss-server-config=httpServerConfigName ,ocss-service-
address={ipAddr1:port1;
      httpClientConfigName} {ipAddr2:port2; httpClientConfigName}
{ipAddr3:port3; httpClientConfigName3}
```

   Configured Example

```
ocss-service-address={192.168.100.105:8060;httpClientConf}
{192.168.100.106:8050;httpClientConf}
```

```
{192.168.100.107:8000;httpClientConf2},ocss-server-config=httpServerConf
```

Next steps

- Log on to theSecurity Shield Dashboard and customize the threat protection settings. See "Customize theSecurity Shield Autonomous Threat Protection Settings" in the *Security Shield User's Guide*.
- Log on to the Security Shield Dashboard and create managed phone number lists. See the "Add Phone Numbers to Policy Rules" in the *Security Shield User's Guide*.

Support for On-Premises Resilience and High Availability

You can configure the Session Border Controller (SBC) to connect to as many as three Cloud Communication Service (CCS) instances simultaneously to provide resilience and continuity of service when one or more CCS instances stops responding or the SBC loses connectivity to CCS or the Oracle® Communications Security Shield Cloud Service (Security Shield).

Connectivity

The SBC tests connectivity all the way to the Security Shield application by sending a "connection check" message to Security Shield through each of the CCS configured instances at one-second intervals. The SBC allows a maximum of 500ms (round trip) for the response from Security Shield before marking the check test as unsuccessful. After three consecutive unsuccessful connection check tests, Security Shield removes the CCS from use for lookup requests. Security Shield continues sending connection check requests to the removed CCS at the specified interval. When Security Shield receives a successful connection check response, the SBC returns the CCS the pool of available CCS instances.

Lookup Requests

Security Shield selects only one CCS for lookup requests, even when more than one CCS is available. For every lookup request, using the CCS configuration order as the prioritized search order, the SBC searches for the first CCS in service and chooses it for sending the request. The SBC always attempts to use the highest priority CCS instance for requests. As CCS instances move in and out of service, there may be transient periods where the SBC sends requests to different CCS instances in the SBC configuration.

> **Note:**
>
> The SBC uses the "hunt" strategy for selection based on the order in which you configured the CCS instances in the SBC configuration.

SBC to CCS Registrations

The SBC registers with only one CCS at a time. Using the same priority list used for lookup requests, the SBC registers with the highest priority CCS available. When the CCS registered to the SBC does not pass the connection check and is taken out of service, the SBC hunts for another CCS. As seen in the Device Status tile on the Security Shield Dashboard, the SBC registration will move from one CCS to another and its registration time will update. Each SBC registration is independent of any other SBC. It is possible that even with identical prioritization of CCS instances in their configurations, SBCs in your network may temporarily register with different CCS instances.

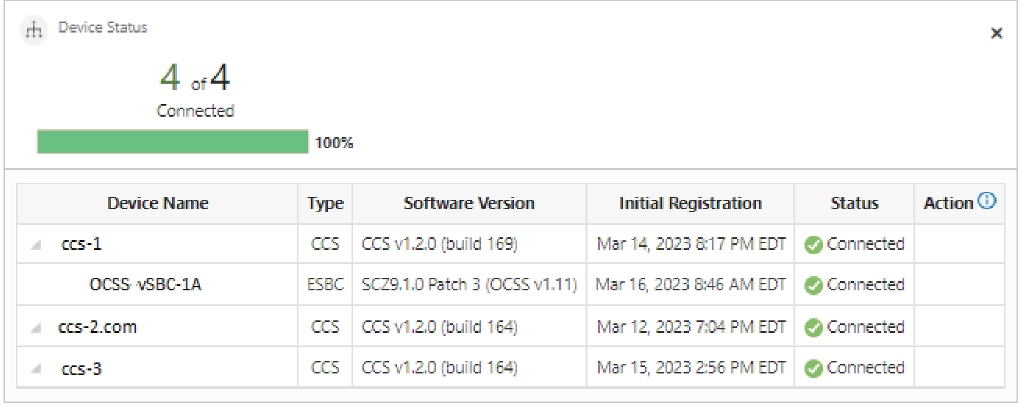
CCS Registration

The CCS registers directly with Security Shield and reports any registered SBCs. Upon completion of the initial registration with Security Shield, the CCS sends periodic registration updates at ten second intervals with one exception. The exception occurs when there are changes to an SBC registration, which results in an immediate CCS registration update to Security Shield.

When registered, the CCS registration time does not update on the Security Shield Device Status tile unless the registration expires. If the registration expires because Security Shield does not receive a registration refresh, the next registration request received by Security Shield results in updating the registration time with the current time.

Device Status and the Activity Log

On the Security Shield Dashboard, the Device Status tile refreshes the registration status of devices at ten second intervals. The following example of the Device Status tile shows the types of information provided.



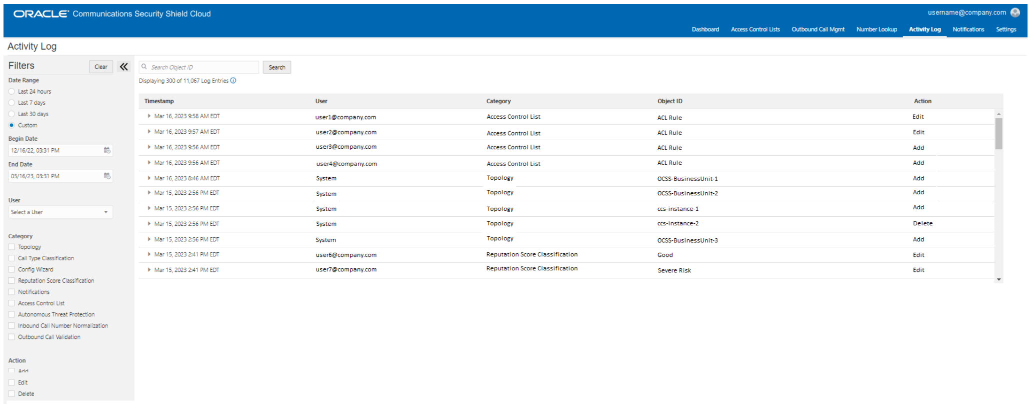
Device Status

4 of 4
Connected

100%

Device Name	Type	Software Version	Initial Registration	Status	Action ⓘ
ccs-1	CCS	CCS v1.2.0 (build 169)	Mar 14, 2023 8:17 PM EDT	✓ Connected	
OCSS vSBC-1A	ESBC	SCZ9.1.0 Patch 3 (OCSS v1.11)	Mar 16, 2023 8:46 AM EDT	✓ Connected	
ccs-2.com	CCS	CCS v1.2.0 (build 164)	Mar 12, 2023 7:04 PM EDT	✓ Connected	
ccs-3	CCS	CCS v1.2.0 (build 164)	Mar 15, 2023 2:56 PM EDT	✓ Connected	

Be aware that although changes are occurring at the SBC and CCS instance, it is possible that the changes do not appear in the Device Status tile at the moment you view the tile due to the ten second refresh rate. Topology changes, even those that revert quickly, for example in less than one second, will display in the activity log. The following example of the Activity Log shows topology changes noted in the Category column. The Object ID and Action columns display additional information about the topology changes.



Activity Log

Filters

Date Range
○ Last 24 hours
○ Last 7 days
○ Last 30 days
● Custom

Begin Date
12/16/22, 03:31 PM

End Date
03/16/23, 03:31 PM

User
Select a User

Category
□ Topology
□ Call Type Classification
□ Config Wizard
□ Reputation Score Classification
□ Notifications
□ Access Control List
□ Autonomous Threat Protection
□ Inbound Call Number Normalization
□ Outbound Call Validation

Action
□ Add
□ Edit
□ Delete

Search Object ID

Displaying 300 of 11,067 Log Entries ⓘ

Timestamp	User	Category	Object ID	Action
Mar 16, 2023 9:58 AM EDT	user1@company.com	Access Control List	ACL Rule	Edit
Mar 16, 2023 9:57 AM EDT	user2@company.com	Access Control List	ACL Rule	Edit
Mar 16, 2023 9:56 AM EDT	user3@company.com	Access Control List	ACL Rule	Add
Mar 16, 2023 9:56 AM EDT	user4@company.com	Access Control List	ACL Rule	Add
Mar 16, 2023 8:46 AM EDT	System	Topology	OCSS-BusinessUnit-1	Add
Mar 15, 2023 2:56 PM EDT	System	Topology	OCSS-BusinessUnit-2	Add
Mar 15, 2023 2:56 PM EDT	System	Topology	ccs-instance-1	Add
Mar 15, 2023 2:56 PM EDT	System	Topology	ccs-instance-2	Delete
Mar 15, 2023 2:56 PM EDT	System	Topology	OCSS-BusinessUnit-3	Add
Mar 15, 2023 2:41 PM EDT	user6@company.com	Reputation Score Classification	Good	Edit
Mar 15, 2023 2:41 PM EDT	user7@company.com	Reputation Score Classification	Severe Risk	Edit

Configuration

To configure connections to the CCS, you must specify the `ocss-service-address` for the CCS instances you want connected to the SBC. From either the ACLI or the Web GUI, go to Configuration, Session Router, `sip-config`, `spl-options`. In the `spl-options` field, enter the `ocss-service-address` for each CCS instance.

You can configure an HTTPS Client for each instance, which allows you to use different clients for connectivity with each CCS including specific certificates for each CCS.

The following example shows the configuration for supporting three CCS instances, where each instance is enclosed in curly brackets. Note that 192.168.100.105;httpClientConf is the CCS LAN side IP address that the Session Border Controller will try to connect to using the HTTP client configuration "httpClientConf".

```
ocss-service-address={192.168.100.105:8060;httpClientConf}
{192.168.100.106:8050;httpClientConf}
{192.168.100.107:8000;httpClientConf2},ocss-server-config=httpServerConf
```

4

Post Service Activation Configuration Tasks

After you activate the Oracle® Communications Security Shield Cloud Service (Security Shield), you might want to configure certain system-wide behaviors through your Oracle Cloud Infrastructure (OCI) Identity Domain account before configuring Security Shield for call traffic. For example, you might want to configure user groups or enable multi-factor authentication. You might also want to configure the Oracle Communications Session Router.

Topics:

- User Groups and Privileges
- Secure Access to Security Shield with Multi-Factor Authentication
- Federated Sign-on
- Session Router Support for Security Shield

User Groups and Privileges

The Oracle® Communications Security Shield Cloud Service (Security Shield) provides a set of user groups to help you manage access to the service according to the least amount of privilege needed. The privileges of each group determine which tabs, links, and information the user can see and which actions the user can perform.

When a user's job requires more privileges than a particular user group allows, the Administrator can assign the user to more groups to provide the right set of privileges for the user's job. For example, suppose a user needs to monitor activity on the system by other users, as well as, to monitor the system. The Administrator can assign the user to both the Security Shield User Tracking and Monitor group and the Security Shield Device Configuration Editor group to give the user the privileges needed to do the job.

User groups are a collection of specific privileges, not user roles. You can use already established user roles, or create new user roles and determine which user groups a role needs. In this way, you can create defined roles and associated privilege needs based on user groups.

Security Shield User and Administrator Groups and Privileges

The following table lists the Security Shield user groups and their privileges.

Groups	Privileges
OCSS ACL Editor—Manages the Access Control Lists, including adding, editing, and deleting lists as well as individual entries.	• Sees the Landing Page and Access Control List (ACL) tabs. • Can view the Detected Threats tile.
CGBU OCSS Administrator—Manages other aspects of the OCCSC service.	• Access the Landing Page and Settings tabs. • Manage on-premises devices. • Access the CCS Configuration and Configuration Wizard on the Settings tab.

Groups	Privileges
OCSS Device Configuration Editor—Manages device configuration.	- Access the Landing Page and Settings tabs. - View the Detected Threats tile. - Manage on-premises devices. - Access the CCS Configuration on the Settings tab.
OCSS Configuration Editor—Manages configuration parameters including thresholds and enforcement actions.	- Access the Landing Page and Settings tabs. - Access to the Autonomous Threat Protection and Configuration Wizard links under Edit Settings. - Access the Security Shield configuration through the Settings tab and modify the configuration. - Access the Configuration Wizard from the Settings tab. - Initiate the Configuration Wizard.
OCSSC User—Monitors call patterns and threats patterns.	- Access the Landing Page tab. - View the Detected Threats tile.
OCSSC User Tracking and Monitoring Editor—Views and manages Activity Logging.	- Access the Landing Page and Activity Log tabs. - View the Detected Threats tile.

For more information about Administrator roles, see Understanding Administrator Roles.

Security Shield Analytics Groups

The following table lists the Security Shield data visualization and analytics groups and their privileges.

Groups	Privileges
OCSSAnalyticsUser—Views the analytics reports.	- View all reports and visualizations (Read-Only).
OCSSAnalyticsEditor—Views and manages the analytics reports for a tenant.	- View all reports and visualizations. - Create, modify, export, and delete reports.

Upgrade and Downgrade Support

Upgrade—Security Shield does not assign any preexisting user accounts to any of the new default groups upon upgrade.

Downgrade—Security Shield allows all user accounts to survive a downgrade and revert to their previous authentication and authorization behavior.

For more information about managing users:

- See User and Role Maintenance, if you use Oracle Identity Cloud Services (IDCS).
- See Managing Users, if you use Oracle Cloud Infrastructure (OCI) Identity Access Management (IAM).

Secure Access to Security Shield with Multi-Factor Authentication

To make the Oracle® Communications Security Shield Cloud Service (Security Shield) more secure, you can enable multi-factor authentication for log on. Multi-factor authentication requires users to provide an additional verification factor for each log on attempt. Users must provide something they know, such as their user name and password, plus something they have, such as a one-time pass-code. With mullti-factor authentication enabled, Security Shield sends a one-time pass-code to the user's email address during the log on attempt. The user must enter the one-time passcode along with user name and password to successfully log on.

See Add a Sign-On Policy.

Federated Sign-on

Federated Sign-on allows you to use a centralized Identity Provider for authenticating users into Oracle® Communications Security Shield Cloud Service (Security Shield). Using a centralized Identity Provider can help you manage all of your user identities from a single source.

You can use Federated Sign-on Security Shield by way of:

- An on-premises Identity and Access Management system
- An Identity Provider that you already use
- Microsoft Active Directory in Azure

See Federating with Identity Providers.

5
Security Shield Maintenance

The Oracle® Communications Security Shield Cloud Service (Security Shield) provides you with tools to monitor and manage your deployment. You can view information about Cloud Communication Service activities in your deployment, change or update certificates, run scripts to change, deactivate, reinstall, upgrade, and downgrade the components, and run show commands.

Topics:

- Cloud Communication Service Metrics, Events, Alarms, and Logs
- Cloud Communication Service Management
- Security Shield Show Commands

Cloud Communication Service Metrics, Events, Alarms, and Logs

The Cloud Communication Service (CCS) can help you monitor its operations and your applications traffic by providing metrics, events, alarms, and logs.

Topics:

- Cloud Communication Service Logs
- Cloud Communication Service Metrics
- Cloud Communication Service Events
- Cloud Communication Service Alarms

Cloud Communication Service Logs

The Cloud Communication Service (CCS) provides logs to help you monitor the health of the service.

The CCS log types include the following:

- NET—network
- DBG—debug
- INF—info
- ERR—error
- EVT—event

The following examples show the format of a log record.

- <date> <timestamp> <thread> <type> <details>
- <date> <timestamp> <thread> <type> <function> <details>

A log mask is passed on startup by way of the command line) with a default of:

- 0001 1100b (EVT+ ERR+ INF+ DBG- NET-).

The CCS writes logs to the console (std::clog).

Logs persist in rotating files with the following defaults:

- 10 files of ~1Mb (required file system space: ~10Mb)
- Filenames are <path>/ccs<0-9>.log where <path> is configured (0 is latest log, 9 is oldest log)

Log Examples

The following is an example of a log file:

```
2020-04-08 10:58:51.652 (0x7fada7b0d240) EVT: CCS
        v1.0.0 (build 0)2020-04-08 10:58:51.652 (0x7fada7b0d240) INF:
main() limits...
2020-04-08 10:58:51.652 (0x7fada7b0d240) INF: main()
parsing...2020-04-08 10:58:51.668 (0x7fada7b0d240) INF: main()
spawning...
2020-04-08 10:58:51.668 (0x7fada7b0d240) INF: Base::Shard::Shard()
shard=0
2020-04-08 10:58:51.669 (0x7fada2dff700) INF: Http11::Client::Client()
client=WAN/idcs.oraclecloud.com:443/oauth2 on shard=0
2020-04-08 10:58:51.669 (0x7fada2dff700) INF: Http20::Client::Client()
client=WAN/ocss.oraclecloud.com:443/ocss on shard=0
2020-04-08 10:58:51.669 (0x7fada2dff700) INF: Http20::Client::Client()
client=WAN/osdmc.oraclecloud.com:443/osdmc on shard=0
2020-04-08 10:58:51.675 (0x7fada2dff700) INF:
Http11::Server::Server()server=WAN/0.0.0.0:9000 on shard=0
2020-04-08 10:58:51.679 (0x7fada2dff700) INF:
Http20::Server::Server()server=LAN/0.0.0.0:8000 on shard=0
2020-04-08 10:58:51.683 (0x7fada2dff700) INF:
Http20::Server::Server()server=OAM/0.0.0.0:2000 on shard=0
2020-04-08 10:58:51.683 (0x7fada2dff700) EVT:
event=WanAuthClientToken/Fsm state=Idle->NoToken service=ocss
2020-04-08 10:58:51.683 (0x7fada2dff700) EVT: alarm=WanAuthClientToken/
Impaired state=CL->CR cause=no auth token service=ocss
2020-04-08 10:58:51.684 (0x7fada2dff700) EVT:
event=WanAuthClientToken/Fsm state=Idle->NoToken service=osdmc
2020-04-08 10:58:51.684(0x7fada2dff700) EVT: alarm=WanAuthClientToken/
Impaired state=CL->CR cause=no auth token service=osdmc
2020-04-08 10:58:51.684(0x7fada2dff700) EVT:
event=WanAuthClientKey/Fsm state=Idle->NoKey
2020-04-08 10:58:51.684 (0x7fada2dff700) EVT: alarm=WanAuthClientKey/
Impaired state=CL->CR cause=no auth key
2020-04-08 10:58:51.688 (0x7fada2dff700) EVT: event=HttpClient/
SessionError shard=0 client=WAN/idcs.oraclecloud.com:443/oauth2
host=unkownn op=net::ip::tcp::resolver::async_resolve() err=Host not
found (authoritative)
2020-04-08 10:58:51.690 0x7fada2dff700) EVT: event=RegServer/Fsm
state=Idle->Active service=ocss
2020-04-08 10:58:51.690 (0x7fada2dff700) EVT: event=RegServer/Fsm
state=Idle->Active service=osdmc
```

```
2020-04-08 10:58:51.691 (0x7fada2dff700) EVT: event=RegClient/Fsm state=Idle-
>Post service=ocss
2020-04-08 10:58:51.691 (0x7fada2dff700) EVT: alarm=RegClient/Isolated
state=CL->CR service=ocss
2020-04-08 10:58:51.691 (0x7fada2dff700) EVT: event=HttpClient/SessionError
shard=0 client=WAN/ocss.oraclecloud.com:443/ocss host=unkownn
op=net::ip::tcp::resolver::async_resolve() err=Host not found(authoritative)
2020-04-08 10:58:51.691 (0x7fada2dff700) EVT: event=RegClient/Fsm state=Idle-
>Post service=osdmc
2020-04-08 10:58:51.691 (0x7fada2dff700) EVT: alarm=RegClient/Isolated
state=CL->CR service=osdmc
2020-04-08 10:58:51.691 (0x7fada2dff700) EVT: event=HttpClient/SessionError
shard=0 client=WAN/osdmc.oraclecloud.com:443/osdmc host=unkownn
op=net::ip::tcp::resolver::async_resolve() err=Host not found(authoritative)
2020-04-08 10:58:51.692 (0x7fada2dff700) EVT: event=OamServer/Fsm state=Idle-
>Active
```

The following example shows a stat log.

```
System/CpuUsage 0 0 0 0
System/MemUsage 0 0 0 0
OamServer/RxGet 0
OamServer/TxError 0

> LAN service "ocss"
RegServer/RxGet 0
RegServer/RxPost 0
RegServer/RxPut 0
RegServer/RxDelete 0
RegServer/TxError 0

> LAN service "osdmc"
RegServer/RxGet 0
RegServer/RxPost 0
RegServer/RxPut 0
RegServer/RxDelete 0
RegServer/TxError 0

> WAN service "ocss"
RegClient/TxPost 3
RegClient/TxPut 0
RegClient/TxDelete 0
RegClient/RxError 2

> WAN service "osdmc"
RegClient/TxPost 3
RegClient/TxPut 0
RegClient/TxDelete 0
RegClient/RxError 2
LanAuth/Failed 0
WanAuth/Failed 0
WanAuthClientKey/TxPost 0
WanAuthClientKey/RxError 0
```

```
> WAN service "ocss"
WanAuthClientToken/TxPost 2
WanAuthClientToken/RxError 1

> WAN service "osdmc"
WanAuthClientToken/TxPost 2
WanAuthClientToken/RxError 1

> Shard 0 Host LAN/0.0.0.0:8000
HttpServer/Sessions 0 0 0 0
HttpServer/RxReq 0
HttpServer/RxReqRate 0 0 0 0
HttpServer/RxReqSize 0 0 0
HttpServer/TxRsp 0
HttpServer/TxRspSize 0 0 0

> Shard 0 Host OAM/0.0.0.0:2000
HttpServer/Sessions 0 0 0 0
HttpServer/RxReq 0
HttpServer/RxReqRate 0 0 0 0
HttpServer/RxReqSize 0 0 0
HttpServer/TxRsp 0
HttpServer/TxRspSize 0 0 0

> Shard 0 Host WAN/0.0.0.0:9000
HttpServer/Sessions 0 0 0 0
HttpServer/RxReq 0
HttpServer/RxReqRate 0 0 0 0
HttpServer/RxReqSize 0 0 0
HttpServer/TxRsp 0
HttpServer/TxRspSize 0 0 0

> Shard 0 Peer WAN/idcs.oraclecloud.com:443/oauth2
HttpClient/Sessions 2 1 0 1HttpClient/TxReq 0
HttpClient/TxReqRate 0 0 0 0
HttpClient/TxReqSize 0 0 0
HttpClient/RxRsp 0
HttpClient/RxRspSize 0 0 0
HttpClient/RxRspLatency 0 0 0

> Shard 0 Peer WAN/ocss.oraclecloud.com:443/ocss
HttpClient/Sessions 1 0 0 1
HttpClient/TxReq 0
HttpClient/TxReqRate 0 0 0 0
HttpClient/TxReqSize 0 0 0
HttpClient/RxRsp 0
HttpClient/RxRspSize 0 0 0
HttpClient/RxRspLatency 0 0 0
> Shard 0 Peer WAN/osdmc.oraclecloud.com:443/osdmc
HttpClient/Sessions 1 0 0 1
HttpClient/TxReq 0
HttpClient/TxReqRate 0 0 0 0
HttpClient/TxReqSize 0 0 0
HttpClient/RxRsp 0
```

```
HttpClient/RxRspSize 0 0 0
HttpClient/RxRspLatency 0 0 0
```

Cloud Communication Service Metrics

The Cloud Communication Service (CCS) collects and reports metrics to keep you informed about traffic, system, and authentication activities. The CCS collects metrics every 15 seconds and reports them every 15 minutes.

The CCS metrics types include:

- Count—A cumulative number that can only increase or reset to zero upon a restart. Count provides a value.
- Gauge—A single value that can go up or down. Gauge provides a value, a minimum, maximum, and average.
- Meter—A specialized gauge that represents a per second rate that can arbitrarily go up or down. Meter provides a value, a minimum, maximum, and average.
- Histogram—A summary of observations marked at 50th, 90th, and 99th percentiles. Historic intervals persist as text in rotating log files with the following defaults:
  - 24 hours of 15 minute intervals (96 files with required system space less than 1 MB).
  - The path to both Regular Logs and Stat Logs is /opt/oracle/ccs/log.
  - Filenames are <path>/stat<0-95>.log, where <path> is configured (0 is latest log, 95 is oldest log).

The following table summarizes CCS metrics.

> **Note:**
>
> In the CCS context, "registration" refers to enabling the ground-to-cloud communication path.

Source	ID	Type	Description	Details	Instancing
System	CpuUsage	Gauge	Gauge of CCS process CPU utilization	CCS process CPU utilization (no per thread stats)	Global
System	MemUsage	Gauge	Gauge of CCS process memory utilization	CCS process memory utilization (no per thread stats)	
HttpServer	Sessions	Gauge	Gauge of HTTP server sessions	HTTP server sessions established	Instanced by: • Shard (thread) • Interface (OAM, LAN, WAN)
HttpServer	RxReq	Count	Count of HTTP requests received	HTTP server requests received	

Source	ID	Type	Description	Details	Instancing
HttpServer	RxReqRate	Meter	Gauge of HTTP requests received	HTTP server requests received (requests per second)	
HttpServer	RxReqSize	Histogram	Histogram of HTTP request sizes received	HTTP server request sizes received (bytes)	
HttpServer	TxRsp	Count	Count of HTTP responses transmitted	HTTP server responses transmitted	
HttpServer	TxRspSize	Histogram	Histogram of HTTP response sizes transmitted	HTTP server response sizes transmitted (bytes)	
HttpClient	Sessions	Gauge	Gauge of HTTP client sessions	HTTP client sessions established	Instanced by: • Shard (thread) • Peer (FQDN and port) • Service (OCSSC, OSDMC)
HttpClient	TxReq	Count	Count of HTTP requests transmitted	HTTP client requests transmitted	
HttpClient	TxReqRate	Meter	Gauge of HTTP requests transmitted	HTTP client requests transmitted (requests per second)	
HttpClient	TxReqSize	Histogram	Histogram of HTTP request sizes transmitted	HTTP client request sizes transmitted (bytes)	
HttpClient	RxRsp	Count	Count of HTTP responses received	HTTP client responses received	
HttpClient	RxRspSize	Histogram	Histogram of HTTP response sizes received	HTTP client response sizes received (bytes)	
HttpClient	RxRspLatency	Histogram	Histogram of HTTP response latency	HTTP client response latency (msec)	
OamServer	RxGet	Count	Count of server GET requests	OAM server GET requests processed	Global
OamServer	TxError	Count	Count of server requests that failed	OAM server requests received that failed (error response)	
RegServer	RxGet	Count	Count of server GET requests	Registration server GET requests processed	Instanced by: • Service (OCSSC, OSDMC)

Source	ID	Type	Description	Details	Instancing
RegServer	RxPost	Count	Count of server POST requests	Registration server POST requests processed	
RegServer	RxPut	Count	Count of server PUT requests	Registration server PUT requests processed	
RegServer	RxDelete	Count	Count of server DELETE requests	Registration server DELETE requests processed	
RegServer	TxError	Count	Count of server requests that failed	Registration server requests received that failed (error response)	
RegClient	TxPost	Count	Count of client POST requests	Registration client POST requests generated	Instanced by: • Service (OCSSC, OSDMC)
RegClient	TxPut	Count	Count of client PUT requests	Registration client PUT requests generated	
RegClient	TxDelete	Count	Count of client DELETE requests	Registration client DELETE requests generated	
RegClient	RxError	Count	Count of client requests that failed	Registration client requests transmitted that failed (error response)	
LanAuth	Failed	Count	Count of LAN and OAM authentication failures	LAN and OAM authentication failures (bad API key)	Global
WanAuth	Failed	Count	Count of WAN authentication failures	WAN authentication failures (bad Identity Domain token)	Global
WanAuthClient Key	TxPost	Count	Count of Identity Domain key client POST requests	Identity Domain key client POST requests generated	Global
WanAuthClient Key	RxError	Count	Count of Identity Domain key client requests that failed	Identity Domain key client requests transmitted that failed (error response)	

Source	ID	Type	Description	Details	Instancing
WanAuthClient Token	TxPost	Count	Count of Identity Domain token client POST requests	Identity Domain token client POST requests generated	Instanced by: • Service (OCSSC, OSDMC)
WanAuthClient Token	RxError	Count	Count of Identity Domain token client requests that failed	Identity Domain token client requests transmitted that failed (error response)	

Example - Statistics Log File

The following example shows a sample log file with statistics for the server (line 14) and the client (line 29).

```
System/CpuUsage 0 0 0 0
System/MemUsage 0 0 0 0
Auth/LanAuthFailed 0
Auth/WanAuthFailed 0
RegClient/Post 0
RegClient/Put 1
RegClient/Del 0
RegClient/Error 0
RegServer/Post 0
RegServer/Put 30
RegServer/Del 0
RegServer/Get 0
RegServer/Error 0
0.0.0.0:2000  (this is a server header for its associated stats below,
and there will be a set per server instance)
HttpServer/Sessions 0 0 0 0
HttpServer/RxReq 0
HttpServer/RxReqRate 0 0 0 0
HttpServer/TxRsp 0
0.0.0.0:443
HttpServer/Sessions 0 0 0 0
HttpServer/RxReq 0
HttpServer/RxReqRate 0 0 0 0
HttpServer/TxRsp 0
0.0.0.0:8080
HttpServer/Sessions 1 0 0 3
HttpServer/RxReq 32
HttpServer/RxReqRate 0 0 0 0
HttpServer/TxRsp 32
144.25.17.233:443  (this is a client header for its associated stats
below, and there will be a set per client instance)
HttpClient/Sessions 0 2 0 5
HttpClient/TxReq 1
HttpClient/TxReqRate 0 0 0 0
HttpClient/RxRsp 1
HttpClient/RxRspLatency 0 0 0
2.0.0.2:5808
```

```
HttpClient/Sessions 0 0 0 0
HttpClient/TxReq 0
HttpClient/TxReqRate 0 0 0 0
HttpClient/RxRsp 0
HttpClient/RxRspLatency 0 0 0
```

Cloud Communication Service Events

The Cloud Communication Service (CCS) records the following stateless events for your information, which typically do not require corrective action. The following table summarizes the supported CCS events.

Source	ID	Fields	Description	Details	Instancing
HttpServer	Exhausted	Shard Interface HostAddr Host Port	HTTP server exhausted	HTTP server session pool exhausted	Instanced by: • Shard (thread) • Interface (OAM, LAN, WAN)
HttpServer	SessionError	Shard Interface HostAddr HostPort PeerAddr PeerPort Operation Error	HTTP server session failed	HTTP server session error Cause is error as returned by networking stack (includes TLS)	
HttpClient	SessionError	Shard Interface PeerAddr PeerPort HostAddr HostPort Operation Error	HTTP client session failed	HTTP client session error Cause is error as returned by networking stack (includes TLS)	Instanced by: • Shard (thread) • Peer (FQDN and port) • Service (OCSSC, OSDMC)
OamServer	Fsm	OldState NewState	OAM server FSM state change	OAM server FSM state change	Global
RegServer	Fsm	OldState NewState Service	Registration server FSM state change	Registration server FSM state change	Instanced by: • Service (OCSSC, OSDMC)
RegServer	DeviceCreated	DeviceId Service	Creation of a peer device registration	Registration server created a device registration record	

Source	ID	Fields	Description	Details	Instancing
RegServer	DeviceDeleted	DeviceId Cause Service	Deletion of a peer device registration	Registration server deleted a device registration record Cause is one of requested or expired	
RegClient	Fsm	OldState NewState Service	Registration client FSM state change	Registration client FSM state change	Instanced by: • Service (OCSSC, OSDMC)
LanAuth	Failed	PeerAddr PeerPort	LAN or OAM API authentication failed	LAN or OAM API authentication failed	Global
WanAuth	Failed	PeerAddr PeerPort	WAN API authentication failed	WAN API authentication failed	Global
WanAuthClient Key	Fsm	OldState NewState	Identity Domain key client FSM state change	Identity Domain client key FSM state change	Global
WanAuthClient Token	Fsm	OldState NewState Service	Identity Domain token client FSM state change	Identity Domain client token FSM state change	Instanced by: • Service (OCSSC, OSDMC)

Cloud Communication Service Alarms

The Cloud Communication Service (CCS) provides the following alarms or your information. Unlike events, alarms are stateful, ranked by severity, and typically require corrective action. See "CCS Operations, Administration, and Maintenance Interface" for information about how to view the data.

If the resolution to an alarm is to check the configuration, you can verify CCS attributes by examining them in the /opt/oracle/ccs/cfg/cfg.json file, or by using the Operations, Administration, and Maintenance (OAM) interface to dump the configuration. Configuration issues with Oracle Cloud Infrastructure (OCI), Identity Domains, and Oracle Communications Security Shield (Security Shield) require assistance from Oracle.

The following table summaizes the supported CCS alarms.

Note:

Network issues are out of scope for this guide.

Source	ID	Severity	Fields	Description	Details	Resolution	Instancing
HttpServer	Down	Critical	Shard Interface HostAddr HostPort Error	HTTP server is unavailable	Raised while HTTP server is not listening Cleared when HTTP server is listening	Investigate reported error and correct. Root causes may include the following: CCS misconfig of host, IP, HTTP server port Network outage	Instanced by: • Shard (thread) • Interface (OAM, LAN, WAN)
Reg Client	Isolated	Critical	Service	Registration of CCS pending	Raised while CCS is not registered with the cloud service Cleared when CCS is registered with the cloud service	Determine why CCS is isolated from the cloud service and correct. Root cause may include... CCS misconfig of WAN IP, HTTP server port, TLS CCS misconfig of Identity Domain FQDN, credentials CCS misconfig of OCSS FQDN Identity Domain misconfig Cloud service misconfig Network outage	Instanced by: • Service (OCSSC, OSDMC)

Source	ID	Severity	Fields	Description	Details	Resolution	Instancing
LanAuth	Impaired	Minor	Cause	LAN API and OAM authentication is impaired	Raised while CCS is configured with an invalid API key Cleared when CCS is configured with a valid API key	Determine why CCS is configured with an invalid API key and correct. Root causes may include the following: CCS misconfig of API key	Global
WanAuthClientKey	Impaired	Critical	Cause	WAN API authentication is impaired	Raised while CCS has not acquired an Identity Domain key Cleared when CCS has acquired an Identity Domain key	Determine why CCS is isolated from Identity Domain and correct. Root causes may include the following: CCS misconfig of WAN IP, HTTP server port, TLS CCS misconfig of Identity Domain FQDN, credentials Identity Domain misconfig Network outage	Global

Source	ID	Severity	Fields	Description	Details	Resolution	Instancing
WanAuthClientToken	Impaired	Critical	Cause Service	WAN API authentication is impaired	Raised while CCS has not acquired an Identity Domain token Cleared when CCS has acquired an Identity Domain token	Determine why CCS is isolated from Identity Domain and correct. Root causes may include the following: CCS misconfig of WAN IP, HTTP server port, TLS CCS misconfig of Identity Domain FQDN, credentials Identity Domain misconfig Network outage	Instanced by: • Service (OCSSC OSDMC)

Example - Regular Log File

The following example shows a sample log file with an alarm entry in line 9.

```
2019-07-16 07:44:58.275 (0x7f2f10be3d80) DBG: parsing..
2019-07-16 07:44:58.306 (0x7f2f10be3d80) DBG: configuring...
2019-07-16 07:44:58.309 (0x7f2f10be3d80) DBG: spawning...
2019-07-16 07:44:58.309 (0x7f2f10be3d80) DBG: Appl::Shards::enable()shards=1
2019-07-16 07:44:58.342 (0x7f2f0bbff700) INF: Http::HttpServer::HttpServer()
HTTP/1.1 server on LAN interface (ipAdress:port)
2019-07-16 07:44:58.346 (0x7f2f0bbff700) INF: Http::HttpServer::HttpServer()
HTTP/1.1 server on WAN interface (ipAdress:port)
2019-07-16 07:44:58.346 (0x7f2f0bbff700) INF: Http::HttpClient::HttpClient()
HTTP/1.1 client on WAN interface (icds.<company>.com:port)
2019-07-16 07:44:58.346 (0x7f2f0bbff700) INF: Http::HttpClient::HttpClient()
HTTP/1.1 client on WAN interface (icds.<company>.com:port)
2019-07-16 07:44:58.356 (0x7f2f0bbff700) EVT: alarm=RegClient/Isolated
state=CR
2019-07-16 07:44:58.367 (0x7f2f0bbff700) ERR: Base::Client::connect()
async_connect()failed for fqdn=ocss.<company>.com:port port=<port number>
with ec=Connection refused
2019-07-16 07:51:19.450 (0x7f2f10be3d80) INF: Util::Signal::block() caught
```

```
signal=2
2019-07-16 07:51:19.451 (0x7f2f10be3d80) DBG: shutdown...
```

Cloud Communication Service Management

Oracle provides a set of scripts that you run on the host to install and manage the Cloud Communication Service (CCS). After the initial installation you can run or re-run any of the scripts to further manage your deployment, including the installation scripts if you need to reinstall the service. The following topics describe the operations you can perform after installation.

Topics:

- The Cloud Communication Service Operations, Administration, and Maintenance Interface
- Cloud Communication Service Certificate Management
- Change the Cloud Communication Service Configuration
- Deactivate the Cloud Communication Service
- Downgrade the Cloud Communication Service
- Uninstall the Cloud Communication Service
- Upgrade the Cloud Communication Service

The Cloud Communication Service Operations, Administration, and Maintenance Interface

When you want to view the Cloud Communication Service (CCS) configuration, metrics, and alarms status, the CCS provides a REST API to enable you to get the information. Through the Operations, Administration, and Maintenance (OAM) interface, you can use any programming language capable of sending and receiving HTTP requests to get the information, for example, client URL Request Library (cURL) and Postman. Note that the information is read-only.

When you want to change the configuration, you must log on to the host, deactivate, make the changes, and reactivate because the CCS does not support dynamic configuration. Specify the host address as 0.0.0.0 and the port as 2000. You must provide a configured API key for authentication.

The <cfg>.json configuration file contains the default host address and port number for the OAM server instance, with a default of 127.0.0.1:2020. You must provide the same API Key that you created for the configuration script for authentication to the OAM server.

> **Note:**
>
> If your deployment requires a different host address and port number, your Oracle support representative can help you edit them in the configuration file.

The CCS supports GET operations on the following URI paths.

Path	Description
/	Help summary
/help	Help summary
/host	Host status
/sys	CCS system status
/cfg	CCS configuration dump
/reg	CCS registration dump
/alarms	CCS alarms
/metrics	CCS metrics
/alarms/server	HTTP server alarms
/alarms/client	HTTP client alarms
/metrics/server	HTTP server metrics
/metrics/client	HTTP client metrics
/raw	All metrics in raw format

> **Note:**
>
> All metrics refresh every 15 seconds and the display refreshes every 15 minutes.

Example for / GET

```
$ curl -k -H "Authorization: Bearer <api-key>" https://<IPadddress>/
-----------------------------------------------------------------------------
Oracle Cloud Communications Service, (c) 2020 Oracle
CCS <version> / <timestamp>
-----------------------------------------------------------------------------

Path                            Description

-----------------------------------------------------------------------------

/—This help summary
/help—This help summary
/host—Host status
/sys—CCS system status
/cfg—CCS configuration dump
/reg—CCS registration dump
/alarms—CCS alarms
/metrics—CCS metrics
/alarms/server—HTTP server alarms
/alarms/client—HTTP client alarms
/metrics/server—HTTP server metrics
/metrics/client—HTTP client metrics
/raw—All metrics in raw format
```

Example for /help GET

The following example shows the results of the GET operation for /help.

```
$ curl -k -H "Authorization: Bearer <api-key>" https://<IPadddress>/
-----------------------------------------------------------------------
-----
Oracle Cloud Communications Service, (c) 2020 Oracle
CCS <version> / <timestamp>
-----------------------------------------------------------------------
-----

Path                        Description

-----------------------------------------------------------------------
-----

/—This help summary
/help—This help summary
/host—Host status
/sys—CCS system status
/cfg—CCS configuration dump
/reg—CCS registration dump
/alarms—CCS alarms
/metrics—CCS metrics
/alarms/server—HTTP server alarms
/alarms/client—HTTP client alarms
/metrics/server—HTTP server metrics
/metrics/client—HTTP client metrics
/raw—All metrics in raw format
-----------------------------------------------------------------------
----
```

Example for /host GET

The following example shows the results of the GET operation for /host.

```
$ curl -k -H "Authorization: Bearer <api-key>" https://<IPadddress>/
-----------------------------------------------------------------------
-----
Oracle Cloud Communications Service, (c) 2020 Oracle
CCS <version> / <timestamp>
-----------------------------------------------------------------------
-----

Hostname                        : <hostname>
Uptime                                : 3d 11:04:57

OS Variant                : Linux
OS Release                : 4.1.12-124.27.1.el7uek.x86_64
OS Version                : #2 SMP Mon May 13 08:56:12 PDT 2019

Host Arch : x86_64
```

```
Num CPUs  : 4
Max CPUs  : 4

Load  1m:  0.00    CPU User  :   0.14%    Mem Total: 14400M
Load  5m:  0.08    CPU System:   0.06%    Mem Used :  1196M
Load 15m:  0.36    CPU Idle  :  99.79%    Mem Free : 13203M
----------------------------------------------------------------------------
```

Example for /sys GET

The following example shows the results of the GET operation for /sys.

```
$ curl -k -H "Authorization: Bearer <api-key>" https://<IPadddress>/
----------------------------------------------------------------------------
Oracle Cloud Communications Service, (c) 2020 Oracle
CCS <version> / <timestamp>
----------------------------------------------------------------------------

Version: CCS v1.0.0 (build 0)
Uptime : 0d 00:17:16
Alarms : (MN 0) (MJ 0) (CR 5)

Process: ccs.exe (PID 104550) (CPU 0.00%) (Mem 0.00%)
----------------------------------------------------------------------------
```

Example for /cfg GET

The following example shows the results of the GET operation for /cfg.

The following example shows a log that the system can generate upon request to help Oracle Customer Support personnel see the active configuration, for example, if adjustments are needed.

```
$ curl -k -H "Authorization: Bearer <api-key>" https://<IPadddress>
----------------------------------------------------------------------------
Oracle Cloud Communications Service, (c) 2020 Oracle
CCS <version> / <timestamp>
----------------------------------------------------------------------------

{
    "Version": 1,
    "System": {
        "Shards": 1,
        "Log-Path": "/mnt/log"
    },
    "HTTP": {
        "Trans-Limit": 1000,
        "Req-Size-Limit": 1,
        "Rsp-Size-Limit": 200,
        "Req-Rx-Timeout": 180,
        "Rsp-Rx-Timeout": 30,
        "Server-Session-Pool": 100,
        "Server-Retry-Timeout": 60,
        "Client-Session-Pool": 4,
        "Client-Retry-Timeout": 10,
        "ALPN-Negotiation": true,
        "Stream-Limit": 128,
        "Stream-Init-Window": 1,
        "Session-Init-Window": 10
    },
    "IDCS": {
        "Refresh-Percentage": 90,
        "Retry-Timeout": 30
    },
    "Registration": {
        "Server-Expiration-Timeout": 60,
        "Server-Expiration-Padding": 10,
        "Client-Device-Name": "ccs.<company>.com",
        "Client-Retry-Timeout": 30,
        "Client-Throttle-Timeout": 10
    },
    "OAM": {
        "Server-Addr": "0.0.0.0",
        "Server-Port": 2000
    },
    "LAN": {
        "Server-Addr": "0.0.0.0",
        "Server-Port": 8000,
        "TLS-Cipher-Suite": "ECDHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-
AES256-GCM-SHA384:ECDHE-RSA-AES128-SHA256:DHE-RSA-AES128-GCM-
SHA256:DHE-RSA-AES256-GCM-SHA384",
        "TLS-Server-Cert": "./ssl/lan-cert.pem",
        "TLS-Server-Key": "./ssl/lan-key.pem",
        "TLS-Server-DH": "./ssl/dh2048.pem",
        "TLS-Client-CA-Path": "./ssl/ca",
        "TLS-Client-Verify": true,
        "API-Key-Verify": true
    },
    "WAN": {
        "Server-FQDN": "ccs.<company>.com",
        "Server-Addr": "0.0.0.0",
        "Server-Port": 9000,
        "TLS-Cipher-Suite": "ECDHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-
AES256-GCM-SHA384",
        "TLS-Server-Cert": "./ssl/wan-cert.pem",
        "TLS-Server-Key": "./ssl/wan-key.pem",
        "TLS-Server-DH": "./ssl/dh2048.pem",
        "TLS-Client-CA-Path": "./ssl/ca",
        "TLS-Client-Verify": true,
        "Identity Domain-FQDN": "idcs.oraclecloud.com",
        "Identity Domain-Port": 443,
        "Identity Domain-Tenant-ID": "idcs-tenant-id",
        "Identity Domain-Verify": true
    },
    "Services": [
        {
            "Prefix": "ocss",
            "FQDN": "ocss.oraclecloud.com",
            "Port": 443,
            "Tenant-ID": "ocss-tenant-id",
            "API-Key": "ocss-lan-api-key",
            "API-Key-Alt": "ocss-lan-api-key-alt",
            "Identity Domain-Client-ID": "ocss-idcs-client-id",
            "Identity Domain-Client-Secret": "ocss-idcs-client-secret"
        },
        {
            "Prefix": "osdmc",
            "FQDN": "osdmc.oraclecloud.com",
            "Port": 443,
            "Tenant-ID": "osdmc-tenant-id",
            "API-Key": "osdmc-lan-api-key",
            "API-Key-Alt": "osdmc-lan-api-key-alt",
            "Identity Domain-Client-ID": "osdmc-idcs-client-id",
            "Identity Domain-Client-Secret": "osdmc-idcs-client-secret"
        }
    ]
}
--------------------------------------------------------------------------
```

Example for /reg GET

The following example shows the results of the GET operation for /reg.

The following example reflects the current registration status of on-premises devices using the Cloud Communication Service (CCS). The example shows only the CCS because the Policy Decision Engine has not yet registered with CCS.

```
$ curl -k -H "Authorization: Bearer <api-key>" https://<IPadddress>/
----------------------------------------------------------------------------
Oracle Cloud Communications Service, (c) 2020 Oracle
CCS <version> / <timestamp>
----------------------------------------------------------------------------

> LAN service "ocss"
{
    "name": "ccs.<company>.com",
    "type": "CCS",
    "version": "CCS v1.0.0 (build 0)",
    "httpAddress": "ccs.<company>.com",
    "httpPort": 9000,
    "devices": []
}

> LAN service "osdmc"
{
    "name": "ccs.<company>.com",
    "type": "CCS",
    "version": "CCS v1.0.0 (build 0)",
    "httpAddress": "ccs.<company>.com",
    "httpPort": 9000,
    "devices": []
}

-----------------------------------------------------------------------
----
```

Example for /alarms GET

The following example shows the results of the GET operation for /alarms.

The following example shows alarm types, severity level, and timestamp. The possible severity levels include:

- CL—Clear
- MN—Minor
- MJ—Major
- CR—Critical

```
$ curl -k -H "Authorization: Bearer <api-key>" https://<IPaddress>/
-----------------------------------------------------------------------
-----
Oracle Cloud Communications Service, (c) 2020 Oracle
CCS <version> / <timestamp>
-----------------------------------------------------------------------
-----

        > WAN service "ocss"
  RegClient                            Lvl  Timestamp
  ------------------------------  --------  -----------------------
  Isolated                              CR  2020-04-11 17:42:54.156

> WAN service "osdmc"
  RegClient                            Lvl  Timestamp
  ------------------------------  --------  -----------------------
  Isolated                              CR  2020-04-11 17:42:54.157

  LanAuth                              Lvl  Timestamp
  ------------------------------  --------  -----------------------
  Impaired                              CL  2020-04-11 18:02:01.890

  WanAuthClientKey                     Lvl  Timestamp
  ------------------------------  --------  -----------------------
  Impaired                              CR  2020-04-11 17:42:54.151

> WAN service "ocss"
  WanAuthClientToken                   Lvl  Timestamp
  ------------------------------  --------  -----------------------
  Impaired                              CR  2020-04-11 17:42:54.151

> WAN service "osdmc"
  WanAuthClientToken                   Lvl  Timestamp
  ------------------------------  --------  -----------------------
  Impaired                              CR  2020-04-11
```

```
17:42:54.151

Cl                    2020-01-23 13:11:05.620
```

Example for /metrics GET

The following example shows the results of the GET operation for /metrics.

In the following example, the Avg, Min, and Max column headings correlate to the percentile values that you can see on the Histogram on the Dashboard. (50th, 90th, and 99th percentiles)

```
$ curl -k -H "Authorization: Bearer <api-key>" https://<IPaddress>/
----------------------------------------------------------------------------
Oracle Cloud Communications Service, (c) 2020 Oracle
CCS <version> / <timestamp>
----------------------------------------------------------------------------

        System                               Val   Avg|P50   Min|P90   Max|
P99
  ------------------------------  --------  --------  --------  --------
  CpuUsage                               0         0         0         0
  MemUsage                               0         0         0         0

  OamServer                            Val   Avg|P50   Min|P90   Max|P99
  ------------------------------  --------  --------  --------  --------
  RxGet                                  4         -         -         -
  TxError                                0         -         -         -

> LAN service "ocss"
  RegServer                            Val   Avg|P50   Min|P90   Max|P99
  ------------------------------  --------  --------  --------  --------
  RxGet                                  0         -         -         -
  RxPost                                 0         -         -         -
  RxPut                                  0         -         -         -
  RxDelete                               0         -         -         -
  TxError                                0         -         -         -

> LAN service "osdmc"
  RegServer                            Val   Avg|P50   Min|P90   Max|P99
  ------------------------------  --------  --------  --------  --------
  RxGet                                  0         -         -         -
  RxPost                                 0         -         -         -
  RxPut                                  0         -         -         -
  RxDelete                               0         -         -         -
  TxError                                0         -         -         -

> WAN service "ocss"
  RegClient                            Val   Avg|P50   Min|P90   Max|P99
  ------------------------------  --------  --------  --------
--------
  TxPost                                 4         -         -
-
  TxPut                                  0         -         -
-
  TxDelete                               0         -         -
-
  RxError                                4         -         -
-

> WAN service "osdmc"
  RegClient                            Val   Avg|P50   Min|P90   Max|
P99
  ------------------------------  --------  --------  --------
--------
  TxPost                                 4         -         -
-
  TxPut                                  0         -         -
-
  TxDelete                               0         -         -
-
  RxError                                4         -         -
-

  LanAuth                              Val   Avg|P50   Min|P90   Max|
P99
  ------------------------------  --------  --------  --------
--------
  Failed                                 0         -         -
-

  WanAuth                              Val   Avg|P50   Min|P90   Max|
P99
  ------------------------------  --------  --------  --------
--------
  Failed                                 0         -         -
-

  WanAuthClientKey                     Val   Avg|P50   Min|P90   Max|
P99
  ------------------------------  --------  --------  --------
--------
  TxPost                                 0         -         -
-
  RxError                                0         -         -
-

> WAN service "ocss"
  WanAuthClientToken                   Val   Avg|P50   Min|P90   Max|
P99
  ------------------------------  --------  --------  --------
--------
  TxPost                                 2         -         -
-
```

```
  RxError                                 2         -         -         -

> WAN service "osdmc"
  WanAuthClientToken                    Val   Avg|P50   Min|P90   Max|P99
  ------------------------------  --------  --------  --------  --------
  TxPost                                  2         -         -         -
  RxError                                 2         -         -
-
```

Example for /alarms/server GET

The following example shows the results of the GET operation for /alarms/server.

```
$ curl -k -H "Authorization: Bearer <api-key>" https://<IPaddress>/alarms/
server
------------------------------------------------------------------------------
--
Oracle Cloud Communications Service, (c) 2020 Oracle
CCS v1.0.0 (build 0) /alarms/server @ 2020-04-11 18:04:53.848
------------------------------------------------------------------------------
--
Shard 0 Host LAN/0.0.0.0:8000
  HttpServer                            Lvl  Timestamp
  ------------------------------  --------  -----------------------
  Down                                   CL  2020-04-11 17:42:54.146

Shard 0 Host OAM/0.0.0.0:2000
  HttpServer                            Lvl  Timestamp
  ------------------------------  --------  -----------------------
  Down                                   CL  2020-04-11 17:42:54.150

Shard 0 Host WAN/0.0.0.0:9000
  HttpServer                            Lvl  Timestamp
  ------------------------------  --------  -----------------------
  Down                                   CL  2020-04-11 17:42:54.142
```

Example for /alarms/client GET

The following example shows the results of the GET operation for /alarms/client.

```
$ curl -k -H "Authorization: Bearer <api-key>" https://<IPaddress>/alarms/
client
------------------------------------------------------------------------------
--
Oracle Cloud Communications Service, (c) 2020 Oracle
CCS v1.0.0 (build 0) /alarms/client @ 2020-04-11 18:05:20.337
------------------------------------------------------------------------------
--
None yet defined.
```

Example for /metrics/server GET

The following example shows the results of the GET operation for /metrics/server.

```
$ curl -k -H "Authorization: Bearer <api-key>" https://<IPaddress>/
metrics/server
-----------------------------------------------------------------------
--------
Oracle Cloud Communications Service, (c) 2020 Oracle
CCS v1.0.0 (build 0) /metrics/server @ 2020-04-11 18:05:56.481
-----------------------------------------------------------------------
--------
> Shard 0 Host LAN/0.0.0.0:8000
  HttpServer                           Val   Avg|P50   Min|P90   Max|
P99
  ------------------------------  --------  --------  --------  
--------
  Sessions                               0         0         0
0
  RxReq                                  0         -         -
-
  RxReqRate                              0         0         0
0
  RxReqSize                              -         0         0
0
  TxRsp                                  0         -         -
-
  TxRspSize                              -         0         0
0

> Shard 0 Host OAM/0.0.0.0:2000
  HttpServer                           Val   Avg|P50   Min|P90   Max|
P99
  ------------------------------  --------  --------  --------  
--------
  Sessions                               0         0         0
0
  RxReq                                  9         -         -
-
  RxReqRate                              0         0         0
0
  RxReqSize                              -         1         1
1
  TxRsp                                  9         -         -
-
  TxRspSize                              -      1024      3072
3072

> Shard 0 Host WAN/0.0.0.0:9000
  HttpServer                           Val   Avg|P50   Min|P90   Max|
P99
  ------------------------------  --------  --------  --------  
--------
  Sessions                               0         0         0
```

```
0
  RxReq                                  0         -         -         -
  RxReqRate                              0         0         0         0
  RxReqSize                              -         0         0         0
  TxRsp                                  0         -         -         -
  TxRspSize                              -         0         0         0
```

Example for /metrics/client GET

The following example shows the results of the GET operation for /metrics/client.

```
$ curl -k -H "Authorization: Bearer <api-key>" https://<IPaddress>/metrics/
client
-----------------------------------------------------------------------------
--
Oracle Cloud Communications Service, (c) 2020 Oracle
CCS v1.0.0 (build 0) /metrics/client @ 2020-04-11 18:06:10.228
-----------------------------------------------------------------------------
--
> Shard 0 Peer WAN/idcs.oraclecloud.com:443/oauth2
  HttpClient                           Val   Avg|P50   Min|P90   Max|P99
  ------------------------------  --------  --------  --------  --------
  Sessions                               2         1         0         1
  TxReq                                  0         -         -         -
  TxReqRate                              0         0         0         0
  TxReqSize                              -         0         0         0
  RxRsp                                  0         -         -         -
  RxRspSize                              -         0         0         0
  RxRspLatency                           -         0         0         0

> Shard 0 Peer WAN/ocss.oraclecloud.com:443/ocss
  HttpClient                           Val   Avg|P50   Min|P90   Max|P99
  ------------------------------  --------  --------  --------  --------
  Sessions                               1         0         0         0
  TxReq                                  0         -         -         -
  TxReqRate                              0         0         0         0
  TxReqSize                              -         0         0         0
  RxRsp                                  0         -         -         -
  RxRspSize                              -         0         0         0
  RxRspLatency                           -         0         0         0

> Shard 0 Peer WAN/osdmc.oraclecloud.com:443/osdmc
  HttpClient                           Val   Avg|P50   Min|P90   Max|P99
  ------------------------------  --------  --------  --------  --------
  Sessions                               1         0         0         0
  TxReq                                  0         -         -         -
  TxReqRate                              0         0         0         0
  TxReqSize                              -         0         0         0
  RxRsp                                  0         -         -         -
  RxRspSize                              -         0         0         0
  RxRspLatency                           -         0         0         0
```

Example for /raw GET

The following example shows the results of the GET operation for /raw.

```
$ curl -k -H "Authorization: Bearer <api-key>" https://<IPaddress>/raw
-----------------------------------------------------------------------
--------
Oracle Cloud Communications Service, (c) 2020 Oracle
CCS v1.0.0 (build 0) /raw @ 2020-04-11 18:07:03.175
-----------------------------------------------------------------------
--------
System/CpuUsage 0 0 0 0
System/MemUsage 0 0 0 0

OamServer/RxGet 11
OamServer/TxError 0

> LAN service "ocss"
RegServer/RxGet 0
RegServer/RxPost 0
RegServer/RxPut 0
RegServer/RxDelete 0
RegServer/TxError 0

> LAN service "osdmc"
RegServer/RxGet 0
RegServer/RxPost 0
RegServer/RxPut 0
RegServer/RxDelete 0
RegServer/TxError 0

> WAN service "ocss"
RegClient/TxPost 14
RegClient/TxPut 0
RegClient/TxDelete 0
RegClient/RxError 14

> WAN service "osdmc"
RegClient/TxPost 14
RegClient/TxPut 0
RegClient/TxDelete 0
RegClient/RxError 14

LanAuth/Failed 0

WanAuth/Failed 0

WanAuthClientKey/TxPost 0
WanAuthClientKey/RxError 0

> WAN service "ocss"
WanAuthClientToken/TxPost 7
WanAuthClientToken/RxError 7
```

```
> WAN service "osdmc"
WanAuthClientToken/TxPost 7
WanAuthClientToken/RxError 7

> Shard 0 Host LAN/0.0.0.0:8000
HttpServer/Sessions 0 0 0 0
HttpServer/RxReq 0
HttpServer/RxReqRate 0 0 0 0
HttpServer/RxReqSize 0 0 0
HttpServer/TxRsp 0
HttpServer/TxRspSize 0 0 0

> Shard 0 Host OAM/0.0.0.0:2000
HttpServer/Sessions 0 0 0 0
HttpServer/RxReq 11
HttpServer/RxReqRate 0 0 0 0
HttpServer/RxReqSize 1 1 1
HttpServer/TxRsp 11
HttpServer/TxRspSize 1024 2432 3072

> Shard 0 Host WAN/0.0.0.0:9000
HttpServer/Sessions 0 0 0 0
HttpServer/RxReq 0
HttpServer/RxReqRate 0 0 0 0
HttpServer/RxReqSize 0 0 0
HttpServer/TxRsp 0
HttpServer/TxRspSize 0 0 0

> Shard 0 Peer WAN/idcs.<company>.com:443/oauth2
HttpClient/Sessions 2 1 0 1
HttpClient/TxReq 0
HttpClient/TxReqRate 0 0 0 0
HttpClient/TxReqSize 0 0 0
HttpClient/RxRsp 0
HttpClient/RxRspSize 0 0 0
HttpClient/RxRspLatency 0 0 0

> Shard 0 Peer WAN/ocss.<company>.com:443/ocss
HttpClient/Sessions 1 0 0 0
HttpClient/TxReq 0
HttpClient/TxReqRate 0 0 0 0
HttpClient/TxReqSize 0 0 0
HttpClient/RxRsp 0
HttpClient/RxRspSize 0 0 0
HttpClient/RxRspLatency 0 0 0

> Shard 0 Peer WAN/osdmc.<company>.com:443/osdmc
HttpClient/Sessions 1 0 0 0
HttpClient/TxReq 0
HttpClient/TxReqRate 0 0 0 0
HttpClient/TxReqSize 0 0 0
HttpClient/RxRsp 0
HttpClient/RxRspSize 0 0 0
HttpClient/RxRspLatency 0 0 0
```

Cloud Communication Service Certificate Management

The Cloud Communication Service (CCS) activation script requires you to enter certain information about the authentication credentials that you want CCS to use when communicating to the Session Border Controller (SBC). The CCS uses certificates and keys to authenticate the SBC.

LAN

For the Local Area Network (LAN) interface you must supply a server certificate, a public key, and a signing certificate.

WAN

For the Wide Area Network (WAN) interface you must supply a server certificate and a public key. You do not need to provide the signing certificate for the WAN because the WAN requires a commercial Certificate Authority. Oracle ships CCS with the root signing keys already provided by the commercial Certificate Authorities.

Use the CCS installation procedure to set the certificates that you want to use for the Oracle® Communications Security Shield Cloud Service service. See Install, Configure, and Activate the Cloud Communication Service.

certificate-record

1. Access the **certificate-record** configuration element.

```
ORACLE# configure terminal
ORACLE(configure)# security
ORACLE(security)# certificate-record
ORACLE(certificate-record)#
```

2. Select the **certificate-record** object to edit.

```
ORACLE(certificate-record)# select
        name                           cert01
        country                        US
        state                          MA
        locality                       Burlington
        organization                   Engineering
        unit
        common-name
        key-size                       1024
        alternate-name
        trusted                        enabled
        key-usage-list
                                       digitalSignature
                                       keyEncipherment
        extended-key-usage-list
                                       serverAuth
        options
        last-modified-by               admin@console
        last-modified-date             2013-10-31 12:35:17
ORACLE(certificate-record)#
```

3. Type **done** to save your configuration.
4. UNUSED STEP; DO NOT DELETE

Change the Cloud Communication Service Configuration

When you want to change the IP address, FQDN, or port for the Cloud Communication Service (CCS), use the CCS installation procedure. Then go to sip-configuration > spl-options on the Session Border Controller and change the ocss-service-address to point to the CCS.

- Install, Configure, and Activate the Cloud Communication Service
- Configure the Session Border Controller for the Security Shield Service

Deactivate the Cloud Communication Service

If you want to deactivate the Cloud Communication Service (CCS) installation, for example to add new certificates or migrate to another host, you can do so without uninstalling the CCS. Use the deactivate script to stop the CCS service from running, while leaving the CCS installed on the system for future re-activation.

Prerequisites

- Confirm that CSS is installed and activated. See Install, Configure, and Activate the Cloud Communication Service for confirmation instructions.
- Be aware of consequences that can affect service.
- Ensure that Perl5 is installed on the host.
- Ensure that you have root privileges.

Procedure

1. Log on to the system.
2. At the prompt type: **/opt/oracle/ccs/perl/deactivate.pl**

   The system verifies that an active CCS instance exists and asks if you want to proceed with deactivate.

   ```
   # /opt/oracle/ccs/perl/deactivate.pl
   ------------------------------------------------------
   Oracle Cloud Communications Service, (c) 2019 Oracle
   CCS <build> <version> deactivate.pl @ 2019-09-17 14:02:48
   ------------------------------------------------------
   Checking pre-conditions...
   Ok.
   Proceed with deactivate (y/n) :
   ```

3. At the prompt, type: **y**.

   The system displays the status.

   ```
   Proceed with deactivate (y/n) : y
   Deactivating...
   Success, ccs-<build>.<version> is down and stopped.
   CONTAINER ID  IMAGE       COMMAND     CREATED     STATUS      PORTS
   NAMES
   ```

4. (Optional) At the prompt, type **docker image ls or podman image ls**, and press Enter to confirm.

   The system displays the REPOSITORY, where you can see that the CCS image no longer exists in the list.

Uninstall the Cloud Communication Service

Use the following procedure when you want to remove the Cloud Communication Service (CCS) from the system, for example, when migrating to another host. If you want to re-install CCS after using this procedure, see "Install, Configure, and Activate CCS."

Prerequisites

- Confirm that CCS is installed and deactivated. See "Deactivate the CSS Installation."
- Be aware of consequences that can affect service.
- Ensure that Perl5 is installed on the host.
- Ensure that you have root privileges.

Procedure

1. Log on to the system.
2. At the prompt, type: **/opt/oracle/ccs/perl/uninstall.pl**.

   The system verifies that an active CCS instance exists and asks if you want to proceed with uninstalling.

   ```
   # /opt/oracle/ccs/perl/uninstall.pl
   ------------------------------------------------------
   Oracle Cloud Communications Service, (c) 2019 Oracle
   CCS <build> <version> activate.pl @ <Date> <Time>
   ------------------------------------------------------
   Checking pre-conditions...
   OK.
   Proceed with uninstall (y/n) :
   ```

3. At the prompt, type: **y**.

   The system displays a success message.

   ```
   Proceed with uninstall (y/n) : y
   Uninstalling...
   Success.
   ```

4. (Optional) Confirm that the system uninstalled CCS using the ls command: **ls /opt/oracle/**.

   ```
   # ls /opt/oracle/
   ls : cannot access /opt/oracle/: No such file or directory
   ```

Upgrade the Cloud Communication Service

Use the following procedure to upgrade the Cloud Communication Service (CCS). The upgrade preserves a snapshot of the most recent installation in the /opt/ocss/ccs directory for a future downgrade, if needed. After you upgrade, ensure that CCS works as expected. If not, downgrade immediately. Note that any configuration changes you made between the last upgrade and the downgrade do not persist.

- Confirm that CCS is installed and activated.
- Know the consequences of an upgrade, such as behavior changes. See the Release Documentation.
- Download the CCS archive file (ccs-<version>.<build>.tgz) that you want to upgrade to from My Oracle Support (MOS) or Oracle SaaSOps

1. Log on to the system.
2. Unpack the ccs-<version>.<build>.tgz archive.

   ```
   tar -xvzf ccs-<version>.<build>.tgz
   ```

   The system creates the ccs-<version> directory and copies the unpacked files there in the following directory tree.

   - install.pl
   - upgrade.pl
   - ccs
   - ccs/.build (hidden)
   - ccs/.version (hidden)
   - ccs/api
   - ccs/api/KeyRsp.v1.json
   - ccs/api/RegReq.v1.json
   - ccs/api/RegRspv1.json
   - ccs/api/TokenRsp.v1.json
   - ccs/cfg
   - ccs/cfg.v1.json
   - ccs/img
   - ccs/img/ccs-<version>.<build>.tar
   - ccs/perl/downgrade.pl
   - ccs/log
   - ccs/perl
   - ccs/perl/activate.pl
   - ccs/perl/config.pl
   - ccs/perl/deactivate.pl

- ccs/perl/uninstall.pl
- ccs/ssl
- ccs/ssl/ca
- ccs/ssl/ca/c_rehash
- ccs/ssl/ca/DigiCertGlobalRootCA.cer
- ccs/ssl/ca/DigiCertSHA256GlobalCaG2.cer
- ccs/ssl/ca/DigiCertSHA256GlobalRootG2.cer
- ccs/ssl/ca/DigiCertSHA2SecureServerCA.cer

3. At the prompt, do the following:
   a. Type **cd ccs-<version>**, and press Enter.
   b. Type **ls**
   c. Type **./ upgrade.pl**

```
# cd ccs-<version>
# ls
# ccs install.pl upgrade.pl
```

4. At the prompt, type ./upgrade.pl, and press Enter.

```
# ./upgrade.pl
--------------------------------------------------------------------
-----------
Oracle Cloud Communications Service, (c) 2020 Oracle
CCS <build> <version> upgrade.pl @ 2020-04-11 19:35:50
--------------------------------------------------------------------
-----------
Checking pre-conditions...
Upgrade from ccs-<build.<version> to ccs-<build.<version> is
supported.
Ok.
Use Docker or Podman (d/p) :
Proceed with upgrade (y/n) :
```

5. Type **d for Docker or p for Podman,** type **y**, and press Enter.

```
Proceed with upgrade (y/n) : y
Backup...
Installing...
Importing...
Success
```

Downgrade the Cloud Communication Service

Use the following procedure to downgrade the Cloud Communication Service (CCS).

- Confirm that Security Shield is installed and activated.
- Know the consequences of a downgrade, such as behavior changes and the loss of configuration changes since the last upgrade.

1. Log on to the system.
2. At the prompt, type /opt/oracle/ccs/perl/downgrade.pl, and press Enter.

   ```
   # /opt/oracle/ccs/perl/downgrade.pl
   --------------------------------------------------------------------------
   -----
   Oracle Cloud Communications Service, (c) 2020 Oracle
   CCS <build> <version> downgrade.pl @ 2020-04-11 19:45:50
   --------------------------------------------------------------------------
   -----
   Checking pre-conditions...
   Downgrade from ccs-<build>.<version> to ccs-<build>.<version> is
   supported.
   Ok.
   Proceed with downgrade (y/n) :
   ```

3. Type y, and press Enter.

   ```
   Proceed with downgrade (y/n) : y
   Reverting...
   Success
   ```

Activate Debug in the CCS

When you want to activate Debug for the Cloud Communication Service (CCS), you must deactivate CCS, edit the ccs/perl/activate.pl script, and reactivate CCS.

Editing the activate.pl script means adding the “--mask=<value>” argument after the existing “--cfg=<value>” argument to set the Debug logs you want. For example:

```
system("docker run --detach --network=host --restart=unless-stopped --
volume=$ccs_dir:/mnt
        --name=$img $img_id --cfg=/mnt/cfg/cfg.json --mask=31 > /dev/null
2>&1") ==
        0 or die "error: $!";
```

To find the mask value, choose the log types you want and add their values together. Enter the sum for the mask value. Oracle assigns a numeric value to each log type, as follows.

- NET (1)
- DBG (2)
- INF (4)
- ERR (8)
- EVT (16)

For example, suppose you want to run DBG, only. Enter 2 for the mask value. Suppose you want to run DBG, NET, and ERR. Enter 11 for the mask value. To run all types, enter 31 for the mask value.

Note:

Remove “--mask=<value>” when you are done debugging.

Security Shield Show Commands

The following information describes the show commands available through the Acme Command Line Interface (ACLI) on your Session Border Controller for viewing Session Plug-in Language (SPL) Application statistics about the Oracle® Communications Security Shield Cloud Service (Security Shield).

The statistics reports are divided into groups. You can view all groups at once or you can specify a single group to view.

Note:

You must load an OCSS.pkg created on or after August 1, 2019 to see the following commands.

Show all SPL Application Stats

Syntax: `show spl appstats`

Use the show spl appstats command to see all Security Shield SPL application statistics reports in one display, which includes the following groups.

- ocss—Displays the SPL build informatiuon.
- ocss-http-stats—Displays Client Requests Sent and Client Requests Received
- ocss-http-stats-detailed—Displays Policy Evaluations Requests (POST), Policy Results, Call Updates (PUT), Call Termination Updates (PUT), Registration Requests (POST), Registration Refresh (PUT), Reregistration (DELETE), Mid-Call Updates, and Mid-Call Actions for up to three Session Border Controllers (SBC) configured for Security Shield.
- ocss-connection-check-stats—Displays statistics for Requests Sent, Response 2xx, Response 400, Response 403, Response 404, Response 4xx, Response 5xx, Response Other, Response Timeout, and Response Invalid.
- ocss-policy-response-time—Displays statistics for 0ms-200ms, 201ms to 500ms, 501ms to 1000ms, 1001ms-1500ms, 1501ms-2000ms, 2001ms-2500ms, 2501ms-3000ms, 3001-3500ms, 3501ms-4000ms, and Above 10000ms, Policy requests, and Average Response Time.
- ocss-policy-rtt (round trip time)—Displays statistics for 0ms-200ms, 201ms to 500ms, 501ms to 1000ms, 1001ms-1500ms, 1501ms-2000ms, 2001ms-2500ms, 2501-3000ms, 3001ms-3500ms, 3501ms-4000ms, 4001ms-10000ms, Above 10000ms, Policy requests, and Average Round TripTime.
- ocss-registration-status—Displays Device Name, Device Type, Resource ID, Registration Interval, OCSS Service Address, Registration State, Registered At, and Local Expire.

- spl show sip circuit-breaker ocss-policy—Displays failureThreshold, retryTimePeriod, checked, checkedHalfOpen, errorCount, notSendOpen, errors, state OPEN, nthSendHalfOpen, windowDuration, notSendHalfOpen, okCount, sendClosed, sendHalfOpen, and ratePrevious. Also, displays the State (Active | Available | Unavailable) and Circuit Breaker State (Closed | Open) of up to three SBCs connected to Security Shield through the Cloud Communication Service (CCS). Also displays the IP address of the active server.
- reset spl-stats application—

Show a Specific SPL Stats Group

To view a specific group of statistics, use the `show spl appstats` command with the group name. For example, to view only the `ocss-policy-rtt` report:

```
show spl appstats ocss-policy-rtt
```

Reset the SPL Stats by Group

To reset a specific group of statistics, use the `reset spl-stats application` command with the group name. For example, to reset the `ocss-policy-rtt` report:

```
reset spl-stats application ocss-policy-rtt
```

Use the following commands to reset the Security Shield SPL application statistics reports by group. See "How to Use the ACLI" in the *ACLI Reference Guide* at https://docs.oracle.com/en/industries/communications/, Enterprise Communications, Enterprise Session Border Controller, <latest release>, User Documentation, ACLI Reference Guide.

A

Download the Cloud Communication Service Software from MOS

As an alternative to downloading the Cloud Communication Service (CCS) software from Oracle Software Delivery Cloud (OSDC), you can download the software from My Oracle Support (MOS).

1. Go to https://support.oracle.com and sign in to My Oracle Support.
2. Click the **Patches and Updates** tab.
3. On the **Patch Search** pane, click the **Search** tab and click **Product or Family (Advanced)**.
4. In the **Product** field, type the full product name. For example, Cloud Communication Service.
5. In the **Release** field, type the release number. For example, OCSSC 20.2.0.
6. Select the software that you want to download, and click **Download**.

B
Add the Security Shield SPL Plug-in

To enable the Session Border Controller (SBC) to perform look ups and add the Oracle® Communications Security Shield Cloud Service (Security Shield) service in the call path, you must add the Security Shield SPL plug-in to `SPL Options` in `SPL Config` and select Enable.

Perform the following procedure from the SBC GUI. You can enable or disable theSecurity Shield SPL plug-in per realm. For more information about SPLs, see "Oracle SPL Plug-ins the ACLI" Configuration Guide.

> **Note:**
>
> Oracle designed Security Shield to monitor traffic between trusted and untrusted domains. Enable the Security Shield SPL only on untrusted realms and not on internal trusted realms.

Procedure

1. Download the Cloud Communication Service (CCS) file from Oracle, which includes the OCSS SPL plug-in, and save the file locally.
2. Untar the .tgz file.
3. SFTP the OCSS.pkg file to /code/spl/ on the SBC. For a High Availability (HA) pair, upload the OCSS.pkg file to both the active and the standby systems.
4. On the SBC, go to **Configuration**, **System**, **SPL Config**.

   The system displays the Modify SPL Config page.
5. On the **Modify SPL Config** page, do the following:

SPL Options	Leave this field empty.
Plug-ins	Click **Add**.

6. On the App SPL Config / Plug-ins page, do the following:

State	Confirm that **enable** is selected.
Name	Select **OCSS.pkg** from the drop-down list.

7. Click **OK**.
8. Click **Save**.

   The system saves and activates the configuration.

C

List of Trusted Certificate Authorities

Oracle trusts the following Certificate Authorities.

```
actalisauthenticationrootca [jdk], Sep 22, 2011, trustedCertEntry,
Certificate fingerprint (SHA-256):
55:92:60:84:EC:96:3A:64:B9:6E:2A:BE:01:CE:0B:A8:6A:64:FB:FE:BC:C7:AA:B5:AF:C1:55:B3:7F:
D7:60:66
addtrustexternalca [jdk], May 30, 2000, trustedCertEntry,
Certificate fingerprint (SHA-256):
68:7F:A4:51:38:22:78:FF:F0:C8:B1:1F:8D:43:D5:76:67:1C:6E:B2:BC:EA:B4:13:FB:83:D9:65:D0:
6D:2F:F2
addtrustqualifiedca [jdk], May 30, 2000, trustedCertEntry,
Certificate fingerprint (SHA-256):
80:95:21:08:05:DB:4B:BC:35:5E:44:28:D8:FD:6E:C2:CD:E3:AB:5F:B9:7A:99:42:98:8E:B8:F4:DC:
D0:60:16
affirmtrustcommercialca [jdk], Jan 29, 2010, trustedCertEntry,
Certificate fingerprint (SHA-256):
03:76:AB:1D:54:C5:F9:80:3C:E4:B2:E2:01:A0:EE:7E:EF:7B:57:B6:36:E8:A9:3C:9B:8D:48:60:C9:
6F:5F:A7
affirmtrustnetworkingca [jdk], Jan 29, 2010, trustedCertEntry,
Certificate fingerprint (SHA-256):
0A:81:EC:5A:92:97:77:F1:45:90:4A:F3:8D:5D:50:9F:66:B5:E2:C5:8F:CD:B5:31:05:8B:0E:17:F3:
F0:B4:1B
affirmtrustpremiumca [jdk], Jan 29, 2010, trustedCertEntry,
Certificate fingerprint (SHA-256):
70:A7:3F:7F:37:6B:60:07:42:48:90:45:34:B1:14:82:D5:BF:0E:69:8E:CC:49:8D:F5:25:77:EB:F2:
E9:3B:9A
affirmtrustpremiumeccca [jdk], Jan 29, 2010, trustedCertEntry,
Certificate fingerprint (SHA-256):
BD:71:FD:F6:DA:97:E4:CF:62:D1:64:7A:DD:25:81:B0:7D:79:AD:F8:39:7E:B4:EC:BA:9C:5E:84:88:
82:14:23
amazonrootca1 [jdk], May 26, 2015, trustedCertEntry,
Certificate fingerprint (SHA-256):
8E:CD:E6:88:4F:3D:87:B1:12:5B:A3:1A:C3:FC:B1:3D:70:16:DE:7F:57:CC:90:4F:E1:CB:97:C6:AE:
98:19:6E
amazonrootca2 [jdk], May 26, 2015, trustedCertEntry,
Certificate fingerprint (SHA-256):
1B:A5:B2:AA:8C:65:40:1A:82:96:01:18:F8:0B:EC:4F:62:30:4D:83:CE:C4:71:3A:19:C3:9C:01:1E:
A4:6D:B4
amazonrootca3 [jdk], May 26, 2015, trustedCertEntry,
Certificate fingerprint (SHA-256):
18:CE:6C:FE:7B:F1:4E:60:B2:E3:47:B8:DF:E8:68:CB:31:D0:2E:BB:3A:DA:27:15:69:F5:03:43:B4:
6D:B3:A4
amazonrootca4 [jdk], May 26, 2015, trustedCertEntry,
Certificate fingerprint (SHA-256):
E3:5D:28:41:9E:D0:20:25:CF:A6:90:38:CD:62:39:62:45:8D:A5:C6:95:FB:DE:A3:C2:2B:0B:FB:25:
89:70:92
baltimorecybertrustca [jdk], May 13, 2000, trustedCertEntry,
Certificate fingerprint (SHA-256):
16:AF:57:A9:F6:76:B0:AB:12:60:95:AA:5E:BA:DE:F2:2A:B3:11:19:D6:44:AC:95:CD:4B:93:DB:F3:
F2:6A:EB
buypassclass2ca [jdk], Oct 26, 2010, trustedCertEntry,
Certificate fingerprint (SHA-256):
9A:11:40:25:19:7C:5B:B9:5D:94:E6:3D:55:CD:43:79:08:47:B6:46:B2:3C:DF:11:AD:A4:A0:0E:FF:
```

```
15:FB:48
buypassclass3ca [jdk], Oct 26, 2010, trustedCertEntry,
Certificate fingerprint (SHA-256):
ED:F7:EB:BC:A2:7A:2A:38:4D:38:7B:7D:40:10:C6:66:E2:ED:B4:84:3E:4C:29:B4:AE:1D:5B:
93:32:E6:B2:4D
camerfirmachambersca [jdk], Aug 1, 2008, trustedCertEntry,
Certificate fingerprint (SHA-256):
06:3E:4A:FA:C4:91:DF:D3:32:F3:08:9B:85:42:E9:46:17:D8:93:D7:FE:94:4E:10:A7:93:7E:
E2:9D:96:93:C0
camerfirmachamberscommerceca [jdk], Sep 30, 2003, trustedCertEntry,
Certificate fingerprint (SHA-256):
0C:25:8A:12:A5:67:4A:EF:25:F2:8B:A7:DC:FA:EC:EE:A3:48:E5:41:E6:F5:CC:4E:E6:3B:71:
B3:61:60:6A:C3
camerfirmachambersignca [jdk], Aug 1, 2008, trustedCertEntry,
Certificate fingerprint (SHA-256):
13:63:35:43:93:34:A7:69:80:16:A0:D3:24:DE:72:28:4E:07:9D:7B:52:20:BB:8F:BD:74:78:
16:EE:BE:BA:CA
certumca [jdk], Jun 11, 2002, trustedCertEntry,
Certificate fingerprint (SHA-256):
D8:E0:FE:BC:1D:B2:E3:8D:00:94:0F:37:D2:7D:41:34:4D:99:3E:73:4B:99:D5:65:6D:97:78:
D4:D8:14:36:24
certumtrustednetworkca [jdk], Oct 22, 2008, trustedCertEntry,
Certificate fingerprint (SHA-256):
5C:58:46:8D:55:F5:8E:49:7E:74:39:82:D2:B5:00:10:B6:D1:65:37:4A:CF:83:A7:D4:A3:2D:
B7:68:C4:40:8E
chunghwaepkirootca [jdk], Dec 20, 2004, trustedCertEntry,
Certificate fingerprint (SHA-256):
C0:A6:F4:DC:63:A2:4B:FD:CF:54:EF:2A:6A:08:2A:0A:72:DE:35:80:3E:2F:F5:FF:52:7A:E5:
D8:72:06:DF:D5
comodoaaaca [jdk], Jan 1, 2004, trustedCertEntry,
Certificate fingerprint (SHA-256):
D7:A7:A0:FB:5D:7E:27:31:D7:71:E9:48:4E:BC:DE:F7:1D:5F:0C:3E:0A:29:48:78:2B:C8:3E:
E0:EA:69:9E:F4
comodoeccca [jdk], Mar 6, 2008, trustedCertEntry,
Certificate fingerprint (SHA-256):
17:93:92:7A:06:14:54:97:89:AD:CE:2F:8F:34:F7:F0:B6:6D:0F:3A:E3:A3:B8:4D:21:EC:15:
DB:BA:4F:AD:C7
comodorsaca [jdk], Jan 19, 2010, trustedCertEntry,
Certificate fingerprint (SHA-256):
52:F0:E1:C4:E5:8E:C6:29:29:1B:60:31:7F:07:46:71:B8:5D:7E:A8:0D:5B:07:27:34:63:53:
4B:32:B4:02:34
digicertassuredidg2 [jdk], Aug 1, 2013, trustedCertEntry,
Certificate fingerprint (SHA-256):
7D:05:EB:B6:82:33:9F:8C:94:51:EE:09:4E:EB:FE:FA:79:53:A1:14:ED:B2:F4:49:49:45:2F:
AB:7D:2F:C1:85
digicertassuredidg3 [jdk], Aug 1, 2013, trustedCertEntry,
Certificate fingerprint (SHA-256):
7E:37:CB:8B:4C:47:09:0C:AB:36:55:1B:A6:F4:5D:B8:40:68:0F:BA:16:6A:95:2D:B1:00:71:
7F:43:05:3F:C2
digicertassuredidrootca [jdk], Nov 10, 2006, trustedCertEntry,
Certificate fingerprint (SHA-256):
3E:90:99:B5:01:5E:8F:48:6C:00:BC:EA:9D:11:1E:E7:21:FA:BA:35:5A:89:BC:F1:DF:69:56:
1E:3D:C6:32:5C
digicertglobalrootca [jdk], Nov 10, 2006, trustedCertEntry,
Certificate fingerprint (SHA-256):
43:48:A0:E9:44:4C:78:CB:26:5E:05:8D:5E:89:44:B4:D8:4F:96:62:BD:26:DB:25:7F:89:34:
A4:43:C7:01:61
digicertglobalrootg2 [jdk], Aug 1, 2013, trustedCertEntry,
Certificate fingerprint (SHA-256):
CB:3C:CB:B7:60:31:E5:E0:13:8F:8D:D3:9A:23:F9:DE:47:FF:C3:5E:43:C1:14:4C:EA:27:D4:
6A:5A:B1:CB:5F
```

```
digicertglobalrootg3 [jdk], Aug 1, 2013, trustedCertEntry,
Certificate fingerprint (SHA-256):
31:AD:66:48:F8:10:41:38:C7:38:F3:9E:A4:32:01:33:39:3E:3A:18:CC:02:29:6E:F9:7C:2A:C9:EF:
67:31:D0
digicerthighassuranceevrootca [jdk], Nov 10, 2006, trustedCertEntry,
Certificate fingerprint (SHA-256):
74:31:E5:F4:C3:C1:CE:46:90:77:4F:0B:61:E0:54:40:88:3B:A9:A0:1E:D0:0B:A6:AB:D7:80:6E:D3:
B1:18:CF
digicerttrustedrootg4 [jdk], Aug 1, 2013, trustedCertEntry,
Certificate fingerprint (SHA-256):
55:2F:7B:DC:F1:A7:AF:9E:6C:E6:72:01:7F:4F:12:AB:F7:72:40:C7:8E:76:1A:C2:03:D1:D9:D2:0A:
C8:99:88
dtrustclass3ca2 [jdk], Nov 5, 2009, trustedCertEntry,
Certificate fingerprint (SHA-256):
49:E7:A4:42:AC:F0:EA:62:87:05:00:54:B5:25:64:B6:50:E4:F4:9E:42:E3:48:D6:AA:38:E0:39:E9:
57:B1:C1
dtrustclass3ca2ev [jdk], Nov 5, 2009, trustedCertEntry,
Certificate fingerprint (SHA-256):
EE:C5:49:6B:98:8C:E9:86:25:B9:34:09:2E:EC:29:08:BE:D0:B0:F3:16:C2:D4:73:0C:84:EA:F1:F3:
D3:48:81
entrust2048ca [jdk], Dec 24, 1999, trustedCertEntry,
Certificate fingerprint (SHA-256):
6D:C4:71:72:E0:1C:BC:B0:BF:62:58:0D:89:5F:E2:B8:AC:9A:D4:F8:73:80:1E:0C:10:B9:C8:37:D2:
1E:B1:77
entrustevca [jdk], Nov 28, 2006, trustedCertEntry,
Certificate fingerprint (SHA-256):
73:C1:76:43:4F:1B:C6:D5:AD:F4:5B:0E:76:E7:27:28:7C:8D:E5:76:16:C1:E6:E6:14:1A:2B:2C:BC:
7D:8E:4C
entrustrootcaec1 [jdk], Dec 18, 2012, trustedCertEntry,
Certificate fingerprint (SHA-256):
02:ED:0E:B2:8C:14:DA:45:16:5C:56:67:91:70:0D:64:51:D7:FB:56:F0:B2:AB:1D:3B:8E:B0:70:E5:
6E:DF:F5
entrustrootcag2 [jdk], Jul 7, 2009, trustedCertEntry,
Certificate fingerprint (SHA-256):
43:DF:57:74:B0:3E:7F:EF:5F:E4:0D:93:1A:7B:ED:F1:BB:2E:6B:42:73:8C:4E:6D:38:41:10:3D:3A:
A7:F3:39
entrustrootcag4 [jdk], May 27, 2015, trustedCertEntry,
Certificate fingerprint (SHA-256):
DB:35:17:D1:F6:73:2A:2D:5A:B9:7C:53:3E:C7:07:79:EE:32:70:A6:2F:B4:AC:42:38:37:24:60:E6:
F0:1E:88
geotrustglobalca [jdk], May 21, 2002, trustedCertEntry,
Certificate fingerprint (SHA-256):
FF:85:6A:2D:25:1D:CD:88:D3:66:56:F4:50:12:67:98:CF:AB:AA:DE:40:79:9C:72:2D:E4:D2:B5:DB:
36:A7:3A
geotrustprimaryca [jdk], Nov 27, 2006, trustedCertEntry,
Certificate fingerprint (SHA-256):
37:D5:10:06:C5:12:EA:AB:62:64:21:F1:EC:8C:92:01:3F:C5:F8:2A:E9:8E:E5:33:EB:46:19:B8:DE:
B4:D0:6C
geotrustprimarycag2 [jdk], Nov 5, 2007, trustedCertEntry,
Certificate fingerprint (SHA-256):
5E:DB:7A:C4:3B:82:A0:6A:87:61:E8:D7:BE:49:79:EB:F2:61:1F:7D:D7:9B:F9:1C:1C:6B:56:6A:21:
9E:D7:66
geotrustprimarycag3 [jdk], Apr 2, 2008, trustedCertEntry,
Certificate fingerprint (SHA-256):
B4:78:B8:12:25:0D:F8:78:63:5C:2A:A7:EC:7D:15:5E:AA:62:5E:E8:29:16:E2:CD:29:43:61:88:6C:
D1:FB:D4
geotrustuniversalca [jdk], Mar 4, 2004, trustedCertEntry,
Certificate fingerprint (SHA-256):
A0:45:9B:9F:63:B2:25:59:F5:FA:5D:4C:6D:B3:F9:F7:2F:F1:93:42:03:35:78:F0:73:BF:1D:1B:46:
CB:B9:12
globalsignca [jdk], Sep 1, 1998, trustedCertEntry,
Certificate fingerprint (SHA-256):
EB:D4:10:40:E4:BB:3E:C7:42:C9:E3:81:D3:1E:F2:A4:1A:48:B6:68:5C:96:E7:CE:F3:C1:DF:
6C:D4:33:1C:99
globalsigneccrootcar4 [jdk], Nov 13, 2012, trustedCertEntry,
Certificate fingerprint (SHA-256):
BE:C9:49:11:C2:95:56:76:DB:6C:0A:55:09:86:D7:6E:3B:A0:05:66:7C:44:2C:97:62:B4:FB:
B7:73:DE:22:8C
globalsigneccrootcar5 [jdk], Nov 13, 2012, trustedCertEntry,
Certificate fingerprint (SHA-256):
17:9F:BC:14:8A:3D:D0:0F:D2:4E:A1:34:58:CC:43:BF:A7:F5:9C:81:82:D7:83:A5:13:F6:EB:
EC:10:0C:89:24
globalsignr3ca [jdk], Mar 18, 2009, trustedCertEntry,
Certificate fingerprint (SHA-256):
CB:B5:22:D7:B7:F1:27:AD:6A:01:13:86:5B:DF:1C:D4:10:2E:7D:07:59:AF:63:5A:7C:F4:72:
0D:C9:63:C5:3B
globalsignrootcar6 [jdk], Dec 10, 2014, trustedCertEntry,
Certificate fingerprint (SHA-256):
2C:AB:EA:FE:37:D0:6C:A2:2A:BA:73:91:C0:03:3D:25:98:29:52:C4:53:64:73:49:76:3A:3A:
B5:AD:6C:CF:69
godaddyclass2ca [jdk], Jun 29, 2004, trustedCertEntry,
Certificate fingerprint (SHA-256):
C3:84:6B:F2:4B:9E:93:CA:64:27:4C:0E:C6:7C:1E:CC:5E:02:4F:FC:AC:D2:D7:40:19:35:0E:
81:FE:54:6A:E4
godaddyrootg2ca [jdk], Sep 1, 2009, trustedCertEntry,
Certificate fingerprint (SHA-256):
45:14:0B:32:47:EB:9C:C8:C5:B4:F0:D7:B5:30:91:F7:32:92:08:9E:6E:5A:63:E2:74:9D:D3:
AC:A9:19:8E:DA
haricaeccrootca2015 [jdk], Jul 7, 2015, trustedCertEntry,
Certificate fingerprint (SHA-256):
44:B5:45:AA:8A:25:E6:5A:73:CA:15:DC:27:FC:36:D2:4C:1C:B9:95:3A:06:65:39:B1:15:82:
DC:48:7B:48:33
haricarootca2015 [jdk], Jul 7, 2015, trustedCertEntry,
Certificate fingerprint (SHA-256):
A0:40:92:9A:02:CE:53:B4:AC:F4:F2:FF:C6:98:1C:E4:49:6F:75:5E:6D:45:FE:0B:2A:69:2B:
CD:52:52:3F:36
identrustcommercial [jdk], Jan 16, 2014, trustedCertEntry,
Certificate fingerprint (SHA-256):
5D:56:49:9B:E4:D2:E0:8B:CF:CA:D0:8A:3E:38:72:3D:50:50:3B:DE:70:69:48:E4:2F:55:60:
30:19:E5:28:AE
identrustpublicca [jdk], Jan 16, 2014, trustedCertEntry,
Certificate fingerprint (SHA-256):
30:D0:89:5A:9A:44:8A:26:20:91:63:55:22:D1:F5:20:10:B5:86:7A:CA:E1:2C:78:EF:95:8F:
D4:F4:38:9F:2F
letsencryptisrgx1 [jdk], Jun 4, 2015, trustedCertEntry,
Certificate fingerprint (SHA-256):
96:BC:EC:06:26:49:76:F3:74:60:77:9A:CF:28:C5:A7:CF:E8:A3:C0:AA:E1:1A:8F:FC:EE:05:
C0:BD:DF:08:C6
luxtrustglobalroot2ca [jdk], Mar 5, 2015, trustedCertEntry,
Certificate fingerprint (SHA-256):
54:45:5F:71:29:C2:0B:14:47:C4:18:F9:97:16:8F:24:C5:8F:C5:02:3B:F5:DA:5B:E2:EB:6E:
1D:D8:90:2E:D5
luxtrustglobalrootca [jdk], Mar 17, 2011, trustedCertEntry,
Certificate fingerprint (SHA-256):
A1:B2:DB:EB:64:E7:06:C6:16:9E:3C:41:18:B2:3B:AA:09:01:8A:84:27:66:6D:8B:F0:E2:88:
91:EC:05:19:50
quovadisrootca [jdk], Mar 20, 2001, trustedCertEntry,
Certificate fingerprint (SHA-256):
A4:5E:DE:3B:BB:F0:9C:8A:E1:5C:72:EF:C0:72:68:D6:93:A2:1C:99:6F:D5:1E:67:CA:07:94:
60:FD:6D:88:73
quovadisrootca1g3 [jdk], Jan 12, 2012, trustedCertEntry,
Certificate fingerprint (SHA-256):
```

```
8A:86:6F:D1:B2:76:B5:7E:57:8E:92:1C:65:82:8A:2B:ED:58:E9:F2:F2:88:05:41:34:B7:F1:F4:BF:
C9:CC:74
quovadisrootca2 [jdk], Nov 24, 2006, trustedCertEntry,
Certificate fingerprint (SHA-256):
85:A0:DD:7D:D7:20:AD:B7:FF:05:F8:3D:54:2B:20:9D:C7:FF:45:28:F7:D6:77:B1:83:89:FE:A5:E5:
C4:9E:86
quovadisrootca2g3 [jdk], Jan 13, 2012, trustedCertEntry,
Certificate fingerprint (SHA-256):
8F:E4:FB:0A:F9:3A:4D:0D:67:DB:0B:EB:B2:3E:37:C7:1B:F3:25:DC:BC:DD:24:0E:A0:4D:AF:58:B4:
7E:18:40
quovadisrootca3 [jdk], Nov 25, 2006, trustedCertEntry,
Certificate fingerprint (SHA-256):
18:F1:FC:7F:20:5D:F8:AD:DD:EB:7F:E0:07:DD:57:E3:AF:37:5A:9C:4D:8D:73:54:6B:F4:F1:FE:D1:
E1:8D:35
quovadisrootca3g3 [jdk], Jan 13, 2012, trustedCertEntry,
Certificate fingerprint (SHA-256):
88:EF:81:DE:20:2E:B0:18:45:2E:43:F8:64:72:5C:EA:5F:BD:1F:C2:D9:D2:05:73:07:09:C5:D8:B8:
69:0F:46
secomscrootca1 [jdk], Sep 30, 2003, trustedCertEntry,
Certificate fingerprint (SHA-256):
E7:5E:72:ED:9F:56:0E:EC:6E:B4:80:00:73:A4:3F:C3:AD:19:19:5A:39:22:82:01:78:95:97:4A:99:
02:6B:6C
secomscrootca2 [jdk], May 29, 2009, trustedCertEntry,
Certificate fingerprint (SHA-256):
51:3B:2C:EC:B8:10:D4:CD:E5:DD:85:39:1A:DF:C6:C2:DD:60:D8:7B:B7:36:D2:B5:21:48:4A:A4:7A:
0E:BE:F6
securetrustca [jdk], Nov 8, 2006, trustedCertEntry,
Certificate fingerprint (SHA-256):
F1:C1:B5:0A:E5:A2:0D:D8:03:0E:C9:F6:BC:24:82:3D:D3:67:B5:25:57:59:B4:E7:1B:61:FC:E9:F7:
37:5D:73
sslrooteccca [jdk], Feb 12, 2016, trustedCertEntry,
Certificate fingerprint (SHA-256):
34:17:BB:06:CC:60:07:DA:1B:96:1C:92:0B:8A:B4:CE:3F:AD:82:0E:4A:A3:0B:9A:CB:C4:A7:4E:BD:
CE:BC:65
sslrootevrsaca [jdk], May 31, 2017, trustedCertEntry,
Certificate fingerprint (SHA-256):
2E:7B:F1:6C:C2:24:85:A7:BB:E2:AA:86:96:75:07:61:B0:AE:39:BE:3B:2F:E9:D0:CC:6D:4E:F7:34:
91:42:5C
sslrootrsaca [jdk], Feb 12, 2016, trustedCertEntry,
Certificate fingerprint (SHA-256):
85:66:6A:56:2E:E0:BE:5C:E9:25:C1:D8:89:0A:6F:76:A8:7E:C1:6D:4D:7D:5F:29:EA:74:19:CF:20:
12:3B:69
starfieldclass2ca [jdk], Jun 29, 2004, trustedCertEntry,
Certificate fingerprint (SHA-256):
14:65:FA:20:53:97:B8:76:FA:A6:F0:A9:95:8E:55:90:E4:0F:CC:7F:AA:4F:B7:C2:C8:67:75:21:FB:
5F:B6:58
starfieldrootg2ca [jdk], Sep 1, 2009, trustedCertEntry,
Certificate fingerprint (SHA-256):
2C:E1:CB:0B:F9:D2:F9:E1:02:99:3F:BE:21:51:52:C3:B2:DD:0C:AB:DE:1C:68:E5:31:9B:83:91:54:
DB:B7:F5
starfieldservicesrootg2ca [jdk], Sep 1, 2009, trustedCertEntry,
Certificate fingerprint (SHA-256):
56:8D:69:05:A2:C8:87:08:A4:B3:02:51:90:ED:CF:ED:B1:97:4A:60:6A:13:C6:E5:29:0F:CB:2A:E6:
3E:DA:B5
swisssigngoldg2ca [jdk], Oct 25, 2006, trustedCertEntry,
Certificate fingerprint (SHA-256):
62:DD:0B:E9:B9:F5:0A:16:3E:A0:F8:E7:5C:05:3B:1E:CA:57:EA:55:C8:68:8F:64:7C:68:81:F2:C8:
35:7B:95
swisssignplatinumg2ca [jdk], Oct 25, 2006, trustedCertEntry,
Certificate fingerprint (SHA-256):
3B:22:2E:56:67:11:E9:92:30:0D:C0:B1:5A:B9:47:3D:AF:DE:F8:C8:4D:0C:EF:7D:33:17:B4:C1:82:
```

```
1D:14:36
swisssignsilverg2ca [jdk], Oct 25, 2006, trustedCertEntry,
Certificate fingerprint (SHA-256):
BE:6C:4D:A2:BB:B9:BA:59:B6:F3:93:97:68:37:42:46:C3:C0:05:99:3F:A9:8F:02:0D:1D:ED:
BE:D4:8A:81:D5
teliasonerarootcav1 [jdk], Oct 18, 2007, trustedCertEntry,
Certificate fingerprint (SHA-256):
DD:69:36:FE:21:F8:F0:77:C1:23:A1:A5:21:C1:22:24:F7:22:55:B7:3E:03:A7:26:06:93:E8:
A2:4B:0F:A3:89
thawteprimaryrootca [jdk], Nov 17, 2006, trustedCertEntry,
Certificate fingerprint (SHA-256):
8D:72:2F:81:A9:C1:13:C0:79:1D:F1:36:A2:96:6D:B2:6C:95:0A:97:1D:B4:6B:41:99:F4:EA:
54:B7:8B:FB:9F
thawteprimaryrootcag2 [jdk], Nov 5, 2007, trustedCertEntry,
Certificate fingerprint (SHA-256):
A4:31:0D:50:AF:18:A6:44:71:90:37:2A:86:AF:AF:8B:95:1F:FB:43:1D:83:7F:1E:56:88:B4:
59:71:ED:15:57
thawteprimaryrootcag3 [jdk], Apr 2, 2008, trustedCertEntry,
Certificate fingerprint (SHA-256):
4B:03:F4:58:07:AD:70:F2:1B:FC:2C:AE:71:C9:FD:E4:60:4C:06:4C:F5:FF:B6:86:BA:E5:DB:
AA:D7:FD:D3:4C
ttelesecglobalrootclass2ca [jdk], Oct 1, 2008, trustedCertEntry,
Certificate fingerprint (SHA-256):
91:E2:F5:78:8D:58:10:EB:A7:BA:58:73:7D:E1:54:8A:8E:CA:CD:01:45:98:BC:0B:14:3E:04:
1B:17:05:25:52
ttelesecglobalrootclass3ca [jdk], Oct 1, 2008, trustedCertEntry,
Certificate fingerprint (SHA-256):
FD:73:DA:D3:1C:64:4F:F1:B4:3B:EF:0C:CD:DA:96:71:0B:9C:D9:87:5E:CA:7E:31:70:7A:F3:
E9:6D:52:2B:BD
usertrusteccca [jdk], Feb 1, 2010, trustedCertEntry,
Certificate fingerprint (SHA-256):
4F:F4:60:D5:4B:9C:86:DA:BF:BC:FC:57:12:E0:40:0D:2B:ED:3F:BC:4D:4F:BD:AA:86:E0:6A:
DC:D2:A9:AD:7A
usertrustrsaca [jdk], Feb 1, 2010, trustedCertEntry,
Certificate fingerprint (SHA-256):
E7:93:C9:B0:2F:D8:AA:13:E2:1C:31:22:8A:CC:B0:81:19:64:3B:74:9C:89:89:64:B1:74:6D:
46:C3:D4:CB:D2
utnuserfirstobjectca [jdk], Jul 10, 1999, trustedCertEntry,
Certificate fingerprint (SHA-256):
6F:FF:78:E4:00:A7:0C:11:01:1C:D8:59:77:C4:59:FB:5A:F9:6A:3D:F0:54:08:20:D0:F4:B8:
60:78:75:E5:8F
verisignclass3g3ca [jdk], Oct 1, 1999, trustedCertEntry,
Certificate fingerprint (SHA-256):
EB:04:CF:5E:B1:F3:9A:FA:76:2F:2B:B1:20:F2:96:CB:A5:20:C1:B9:7D:B1:58:95:65:B8:1C:
B9:A1:7B:72:44
verisignclass3g4ca [jdk], Nov 5, 2007, trustedCertEntry,
Certificate fingerprint (SHA-256):
69:DD:D7:EA:90:BB:57:C9:3E:13:5D:C8:5E:A6:FC:D5:48:0B:60:32:39:BD:C4:54:FC:75:8B:
2A:26:CF:7F:79
verisignclass3g5ca [jdk], Nov 8, 2006, trustedCertEntry,
Certificate fingerprint (SHA-256):
9A:CF:AB:7E:43:C8:D8:80:D0:6B:26:2A:94:DE:EE:E4:B4:65:99:89:C3:D0:CA:F1:9B:AF:64:
05:E4:1A:B7:DF
verisignuniversalrootca [jdk], Apr 2, 2008, trustedCertEntry,
Certificate fingerprint (SHA-256):
23:99:56:11:27:A5:71:25:DE:8C:EF:EA:61:0D:DF:2F:A0:78:B5:C8:06:7F:4E:82:82:90:BF:
B8:60:E8:4B:3C
xrampglobalca [jdk], Nov 1, 2004, trustedCertEntry,
Certificate fingerprint (SHA-256):
CE:CD:DC:90:50:99:D8:DA:DF:C5:B1:D2:09:B7:37:CB:E2:C1:8C:FB:2C:10:C0:FF:0B:CF:0D:
32:86:FC:1A:A2
```

D
Changes to IDCS and OCI IAM Operations

Oracle recently merged the Identity Cloud Services (IDCS) operations into the native Oracle Cloud Infrastructure (OCI) and Identity Access Management (IAM) service, no longer offering IDCS as a separate service. The following information describes the changes and what they mean to both IDCS and OCI IAM users.

As of January 17, 2022, new Oracle® Communications Security Shield Cloud Service (Security Shield) customers will manage their tenancies through OCI Identity Domain.

During February 2022, Oracle begins migrating existing IDCS instances to the new OCI Identity Domain model. Existing customers can manage their tenancies through IDCS until their migration completes.

During March, 2022, tenancy management through IDCS ends. All customers manage their Security Shield tenancies through OCI Identity Domain from this date forward.

> **Note:**
>
> The updated service will not be deployed to all regions at once. Banners on the IDCS and OCI sign on screens will indicate when identity domains are enabled in your region and where to find more information.

Topics:

- OCI Identity Domains: What Oracle IDCS Customers Need to Know
- OCI Identity Domains: What OCI Customers Need to Know

OCI Identity Domains: What Oracle IDCS Customers Need to Know

Oracle recently merged the capabilities of Oracle Identity Cloud Service (IDCS) into the native Oracle Cloud Infrastructure Identity and Access Management (OCI IAM) service. As a native OCI service, customers will see improved performance and scale, immediate availability in more global regions, and a new cross-region disaster recovery feature.

What is OCI Identity Domain?

Oracle Cloud Infrastructure (OCI) Identity Domain is the access control plane for Oracle Cloud. An identity domain is a container for managing users and roles, federating and provisioning of users, secure application integration through Oracle Single Sign-On (SSO) configuration, and SAML and OAuth based Identity Provider administration.

For more information about Identity Domains, see IAM with Identity Domains and Managing Identity Domains.

What Changed for IDCS and Identity Domain?

Oracle recently made new features and capabilities available for the Oracle Cloud Infrastructure (OCI) Identity Domain service. As part of the upgraded service, Oracle migrated the features and functionality of the existing Oracle Identity Cloud Service (IDCS) into OCI Identity Domain.

OCI Identity Domain supports the following core functions:

- OCI Identity Domain continues to serve as the critical access control plane for Oracle Cloud.
- OCI Identity Domain supports a wide range of enterprise Identity Domain use cases for complex, hybrid IT environments.
- OCI Identity Domain provides a developer-friendly Identity Domain engine for custom and consumer applications.

By unifying administration and user experiences across key Identity Domain functions, the new service helps simplify administration, reduce cost of ownership, and improve time-to-value. The service spans Cloud and on-premises, providing the flexibility to handle a wide variety of Identity Domain use cases across employee, partner, and consumer scenarios. As a native service of OCI, you can use the diverse feature set of OCI Identity Domain across any geography.

The updated OCI Identity Domain service introduces Identity Domains. Oracle will migrate your existing IDCS instances, called stripes, to Identity Domain instances. Existing Security Shield customer will see their access to IDCS portal diverted to Identity Domain. No changes are required to applications, users, or groups in domains that formerly existed as IDCS instances or to local users in OCI tenancies. See Identity Domains.

- Identity domains are the next generation of IDCS instances. Each existing IDCS instance is now an identity domain.
- Each OCI identity domain represents a stand-alone identity and access management solution.
- Identity domains each have their own settings, configurations, and security policies to ensure optimal security.

How Does The Upgrade to OCI Identity Domain Impact Existing Identity Cloud Service Instances?

None of the existing Oracle Identity Cloud Service (IDCS) features or functionality will change as part of the migration to Oracle Cloud Infrastructure (OCI) Identity Domain. Oracle will merge IDCS into OCI Identity Domain, where it will become an integral component.

As a native service of OCI, OCI Identity Domain takes advantage of infrastructure that offers consistently high performance, enterprise scalability, availability in all the Oracle global cloud regions, and an extensive set of regulatory compliance and security certifications.

The OCI Identity Domain service will serve all current IDCS use cases, including providing a standalone Identity as a Service (IDaaS) solution for managing access across numerous third-party applications. IDCS customers migrating to OCI Identity

Domain do not need to consume any other OCI services to continue using the services previously provided by IDCS.

Oracle will prepare each IDCS instance to be managed through the OCI console as an identity domain. All existing configurations, security settings, user and group populations, and access assignments will continue to exist with no interruption. Users who authenticate through custom sign-on screens may not even know that a change occurred.

The system will re-route IDCS Administrators from the existing IDCS administrative console to the Identity Domain console where IDCS instances will be listed as OCI Identity Domains. Administrators can browse to their list of domains and will be able to manage domains in a way similar to the current IDCS console experience. See Managing Identity Domains.

The upgrade makes no changes to pricing, metering, or included features for Security Shield instances. You will continue to use your existing Security Shield entitlements and any others you are entitled to use.

What is New in OCI Identity Domain for IDCS Customers?

The migration to Oracle Cloud Infrastructure (OCI) Identity Domain and the introduction of identity domains adds Oracle Identity Cloud Service (IDCS) features natively to the OCI Identity Domain service.

- Single-Point of Identity Domain Management—Identity administration is now available through the OCI Admin console under Identity & Security, Domains. Administrators will see the same set of features and functionality that they are used to in IDCS for managing users, groups, applications, security settings, and other configurations.
- No Impact for Existing Users, Policies, Configuration, or Access—The OCI Identity Domain upgrade maintains all existing security policies, configurations, and user populations. Expect no impact to security settings or to the user experience. Oracle did not remove functionality or change any policy configurations.
- Disaster Recovery—In most regions, OCI Identity Domain now provides a cross-region disaster recovery feature for recovering identity domain data in a scenario where an entire OCI region becomes unavailable. The disaster recovery feature is included and does not require any changes or updates to existing applications.

Post-Upgrade Guidance

Administrative Access

Identity Cloud Service (IDCS) Administrators become Identity Domain Administrators upon migration. Identity Domain Administrators get full access to their identity domains. Be sure that use of the OCI Administrators group is consistent with your security policies.

The Oracle® Communications Security Shield Cloud Service (Security Shield) Administrators group grants access to many aspects of the service. Oracle recommends reserving the Security Shield Administrators group for emergency scenarios, rather than for day-to-day administration of the tenancy. Best practices include

- discontinuing the use of the Administrators account after initial setup.
- setting a complex password on the account.
- storing the Administrators account credentials safely in a secure location such as a physical safe.

Where Can I Get More Information?

Use the following resources to find more information about Oracle Cloud Infrastructure (OCI) and Identity Domains.

- IAM with Identity Domains
- oracle.com
- In North America, call +1.800.ORACLE1 (672-2531)
- Outside North America, find your local Oracle office at oracle.com/contact

OCI Identity Domains: What OCI Customers Need to Know

Oracle recently merged the capabilities of Oracle Identity Cloud Service (IDCS) into the native Oracle Cloud Infrastructure (OCI) service. The merger provides OCI customers with a rich, enterprise-class set of identity and access management features for use with OCI and Oracle Cloud applications.

What is OCI Identity Domain?

Oracle Cloud Infrastructure (OCI) Identity Domain is the access control plane for Oracle Cloud. An identity domain is a container for managing users and roles, federating and provisioning of users, secure application integration through Oracle Single Sign-On (SSO) configuration, and SAML and OAuth based Identity Provider administration.

For more information about Identity Domains, see IAM with Identity Domains and Managing Identity Domains.

What Changed for Security Shield?

Oracle recently made new features and capabilities available for the Oracle Cloud Infrastructure Identity (OCI) and Identity Domain service. As part of the upgraded service, Oracle merged all features and functionality of the existing Oracle Identity Cloud Service (IDCS) into OCI Identity Domain.

OCI Identity Domain supports the following core functions:

- OCI Identity Domain continues to serve as the critical access control plane for Oracle® Communications Security Shield Cloud Service (Security Shield).
- OCI Identity Domain supports a wide range of enterprise Identity Domain use cases for complex, hybrid IT environments.
- OCI Identity Domain provides a developer-friendly Identity Domain engine for custom and consumer applications.

Identity Domain is also flexible enough to handle a wide variety of Identity Domain use cases across employee, partner, and consumer scenarios.

The updated OCI Identity Domain service introduces Identity Domains. Oracle will migrate your existing IDCS instances, called stripes, to Identity Domain instances. Existing Security Shield customer will see their access to IDCS portal diverted to Identity Domain. No changes are required to applications, users, or groups in domains that formerly existed as IDCS instances or to local users in OCI tenancies. See Identity Domains.

Identity Domain characteristics include:

- Each OCI Identity Domain represents a stand-alone identity and access management solution.
- Each identity domain represents a different user population, but certain use cases may require users to exist in multiple domains.
- Identity domains each use their own settings, configurations, and security policies to ensure optimal security.
- OCI Identity Domain is an Identity as a Service (IDaaS) solution with the flexibility to cover virtually any Identity Domain use cases across employees, partners, and consumers.

How Do the Changes to OCI IAM Impact Existing OCI Tenancies?

OCI administrators are already be familiar with the Oracle Cloud Infrastructure Identity and Access Management (OCI IAM) service that enables authentication into OCI and management of access entitlements for OCI resources by way of OCI IAM policies. Many customers choose to use Oracle Identity Cloud Service (IDCS) to also enable more advanced IAM deployments, which creates an additional layer of IAM to manage and sometimes incurs additional cost.

The introduction of identity domains adds the following features natively to the OCI IAM service to help simplify administration and operational management.

- Powerful IAM Functionality at No Additional Cost—Oracle brought all the enterprise IAM capabilities of IDCS into OCI IAM natively. IAM functionality such as advanced authentication techniques and user life cycle management are now natively available and included in your existing OCI tenancies for use with your subscribed* Oracle services.

  > **Note:**
  >
  > *Upgrades are available to provide IAM support beyond subscribed Oracle services.

- Single-Point Authentication—The OCI IAM upgrade simplifies the OCI sign-on screen.
- Single-Point of IAM Management—Customers who previously used IDCS with OCI tenancies may notice simplified administration by way of a single pane for all users. Identity administration is now available through the OCI Admin console under Identity & Security, Domains.
- No Impact for Existing Users, Policies, Configuration, or Access—The OCI IAM upgrade maintains all existing security policies, configurations, and user populations. Expect no impact to security settings or to the user experience. Oracle did not remove functionality or change any policy configurations.
- Disaster Recovery—OCI IAM now provides a cross-region disaster recovery feature for recovering identity domain data in a scenario where an entire OCI region becomes unavailable. The disaster recovery feature is included and does not require any changes or updates to existing applications.

Post-Upgrade Guidance

Administrative Access

Identity Cloud Service (IDCS) Administrators become Identity Domain Administrators upon migration. Identity Domain Administrators get full access to their identity domains. Be sure that use of the OCI Administrators group is consistent with your security policies.

The Oracle® Communications Security Shield Cloud Service (Security Shield) Administrators group grants access to many aspects of the service. Oracle recommends reserving the Security Shield Administrators group for emergency scenarios, rather than for day-to-day administration of the tenancy. Best practices include

- discontinuing the use of the Administrators account after initial setup.
- setting a complex password on the account.
- storing the Administrators account credentials safely in a secure location such as a physical safe.

Where Can I Get More Information?

Use the following resources to find more information about Oracle Cloud Infrastructure (OCI) and Identity Domains.

- IAM with Identity Domains
- oracle.com
- In North America, call +1.800.ORACLE1 (672-2531)
- Outside North America, find your local Oracle office at oracle.com/contact