

Oracle® Communications Service Controller

Installation Guide

Release 6.2

F18710-02

April 2020

Copyright © 2010, 2020, Oracle and/or its affiliates.

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

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

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

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software" or "commercial computer software documentation" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

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

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

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

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

Contents

Preface	v
Audience	v
Documentation Accessibility	v
1 About Service Controller Deployments	
Understanding the Domain-Based Administration Model	1-1
About Managed Servers	1-1
About the Domain Configuration Directory	1-2
About the Administration Server	1-2
Domain Types	1-2
Deployment Topologies	1-3
Multi-Domain Topology	1-3
Unified Domain Topology	1-4
About Clusters	1-4
About Setting the Service Availability Mode	1-5
About Production Environments	1-5
About Test and Evaluation Systems	1-6
Scaling the Deployment	1-7
Scaling the Processing Tier	1-7
Scaling the Signaling Tier	1-7
2 System Requirements	
Hardware Requirements	2-1
Managed Servers	2-1
Administration Server	2-1
Software Requirements	2-2
Supported Operating Systems and JDKs	2-2
Supported Web Browsers	2-2
3 Pre-Installation Tasks	
Downloading the Service Controller Software	3-1
Setting Socket Buffer Sizes	3-1

4 Installing the Software

About the Service Controller Installer	4-1
Choosing an Oracle Home	4-1
Choosing an Oracle Central Inventory	4-2
Choosing an Installation Mode	4-2
Installing Service Controller Products in Graphical Mode	4-2
Installing Service Controller Products in Silent Mode	4-3
Installation Parameters in the silentInstall.properties File	4-4
Directory Structure After Installation	4-6
Deinstalling Service Controller	4-7

5 Setting Up a Domain

About Domain Setup	5-1
Understanding the Domain Configuration Directory	5-1
Installing the Administration Server and Managed Servers	5-2
Creating Multiple Managed Servers by Copying Them	5-3
Securing Domains	5-4
Creating a Domain in Interactive Mode	5-4
Creating a Domain in Silent Mode	5-6
Specifying Parameters in an XML File	5-7
Domain Creation Parameters in the create_domain_silent.xml File	5-7
Entering Arguments at the Command Line	5-8
Starting the Domain Web Server	5-8
Configuring Security for the Administration Server	5-9
Starting The Administration Console	5-11
Adding a Managed Server to The Domain	5-11
Starting the Managed Server	5-13

6 Next Steps

Starting the SS7 Process	6-1
Performing Additional Setup Tasks for Service Controller Products	6-1
Completing Security Configuration for the System	6-2
Configuring SNMP	6-2

Preface

This Installation Guide describes how to install Oracle Communications Service Controller and configure a domain and is relevant to all Service Controller products.

Audience

This guide is intended for system engineers, integrators, and administrators who will install and configure Service Controller to support the desired solution architecture.

It is assumed you have advanced knowledge in the following areas: Service Controller concepts, your operator's network, system hardware and operating system. See *Service Controller Concepts Guide* for more information.

Documentation Accessibility

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

Access to Oracle Support

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

About Service Controller Deployments

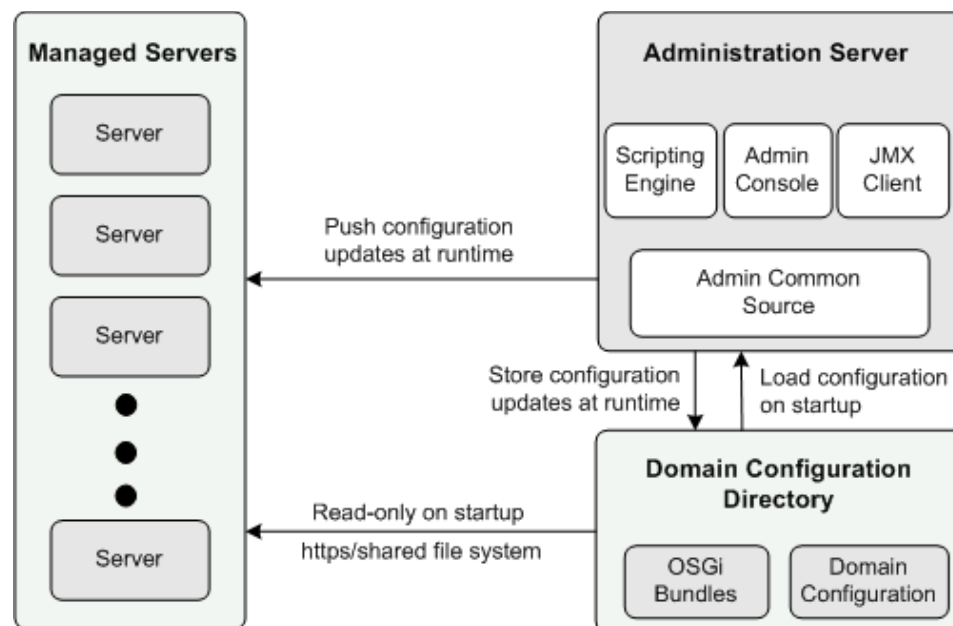
This chapter describes essential concepts for understanding Oracle Communications Service Controller deployments.

Understanding the Domain-Based Administration Model

Service Controller is implemented and administered using the concept of *domains*. A domain is a logically related group of servers.

Figure 1-1 shows domain components, which are explained in the following sections.

Figure 1-1 A Service Controller Domain



About Managed Servers

Each managed server runs on its own Java Virtual Machine (JVM). Managed servers in a domain all have the same Open Services Gateway Initiative (OSGi) software bundles installed and started.

Managed servers have read-only access to the domain configuration directory. At startup, the servers load the software bundles and the required configuration from the domain configuration directory.

Managed servers can be added and removed from the domain without service interruption while the system is running.

About the Domain Configuration Directory

Each domain has one associated domain configuration which is stored in the domain configuration directory. The domain configuration directory contains:

- The configuration details stored in XML files for all the managed servers in the domain.
- The OSGi software bundles that the managed servers run.

When a managed server starts, it retrieves configuration data from the domain configuration directory and loads it into memory. Individual servers do not store configuration locally except for the initial configuration and the security-related details they need to enable access to the domain configuration directory.

Changes in the domain configuration directory are managed by the Administration Server. The Administration Server saves all configuration updates to the domain configuration directory during runtime.

Managed servers access the domain configuration directory using either of these data sharing methods:

- Shared network file system
- Domain Configuration Web server that connects to the managed servers using HTTP or HTTPS.

Note: HTTP should never be used in a production environment.

About the Administration Server

The Administration Server enables you to manage the domain servers, the OSGi software bundles installed in the domain, and the data stored in the domain configuration directory. Oracle recommends using a dedicated computer for the Administration Server.

You can access the Administration Server using these clients:

- Administration Console:
Web access enables administrators to configure the domain from any computer with a Web browser and network access to the Administration Server.

- JConsole or Scripting Engine:

If you want, you can interact programmatically with the Administration Server by using JMX configuration MBeans. Typically, working with MBeans involves integrating Service Controller with a JMX-enabled network management system.

Scripts can be used if you need to repeat lengthy and complicated configuration changes. The scripting engine is a shell script that accepts an XML file argument. The XML file defines operations and attributes on Administration Server configuration MBeans.

Domain Types

The domain type reflects which sets of software bundles are run on the managed servers and which Service Controller roles they perform.

There are three domain types:

- Signaling domain:

Managed servers in the signaling domain run the software bundles associated with the Service Controller signaling tier. These components include the various signaling server units (SSUs) that enable network connectivity. Servers in the signaling domain are often referred to as signaling servers.

- Processing domain:

Managed servers in the processing domain run the software bundle associated with the Service Controller processing tier. These components include the Interworking Modules (IMs), an Orchestration Engine (OE), applications, and mediators that enable traffic processing and mediation functions. Servers in the processing domain are often referred to as processing servers.

- Unified domain:

This domain combines the processing and signaling tier functions. Managed servers in the unified domain run the bundles associated with both the signaling tier and processing tier.

Deployment Topologies

This section describes the topologies you can optionally use for Service Controller deployments. The selection of a deployment topology will affect level of performance, availability, and continuity. See the description of service availability and continuity in the ["About Setting the Service Availability Mode"](#) section for more information.

Multi-Domain Topology

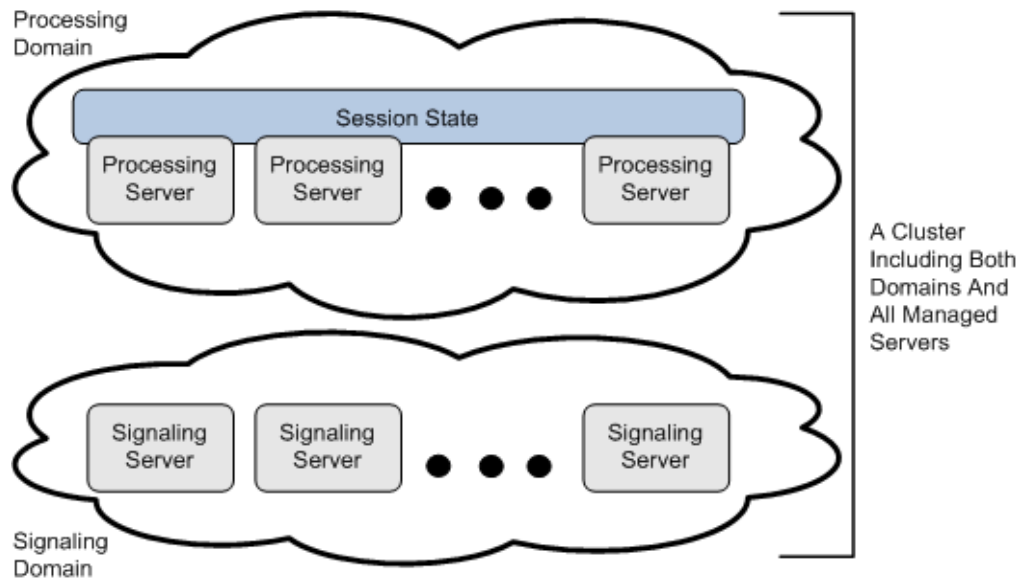
A multi-domain topology implements two distinct types of domains:

- A signaling domain
- A processing domain

State information can be shared across servers in the domain, ensuring service continuity in case of server failure. The signaling domain and processing domain interact by propagating protocol events across the tier and domain boundaries.

[Figure 1–2](#) shows a multi-domain topology with one signaling domain and one processing domain, with the session state distributed across the processing servers.

Figure 1–2 A Multi-Domain Topology



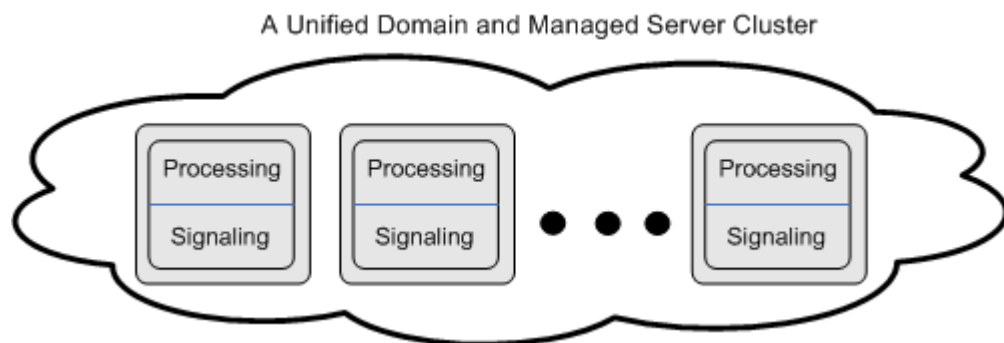
Note: If your telecom system connects to multiple networks, you should consider implementing an additional signaling domain for each network.

Unified Domain Topology

Each managed server in a unified domain performs the role of both the signaling tier and the processing tier. Multiple managed servers in the domain enable server availability following a server failure. See "[About Setting the Service Availability Mode](#)" for more information about availability and continuity options.

Figure 1–3 shows a unified domain topology with one unified domain in which servers perform the role of both the processing and signaling tiers.

Figure 1–3 A Unified Domain Topology



About Clusters

Service Controller creates managed server clusters using the *Oracle Coherence* product technology, which provides replicated and distributed data management and caching

services. Coherence uses a highly scalable peer-to-peer clustering protocol. See the Oracle Coherence product documentation for an introduction to this technology.

Oracle Coherence automatically and transparently fails over and redistributes its clustered data management services. When a new managed server is added, or when a failed managed server is restarted, it automatically joins the cluster.

A Service Controller cluster provides the following additional services:

- Interprocess communication
- Persistence

Managed servers in all domains (processing, signaling, and unified) distribute events among each other across domain boundaries.

To make the domains aware of each other, you need to group them in a cluster using either of these two methods, which are mutually exclusive:

- Using IP multicast: In this case, you need to specify a single IP address. All servers use this address to send and receive broadcast messages. Multicast is the default option when you use the domain creation script.
- Using IP unicast: In this case, you need to specify IP addresses of all servers in both domains.

For more information on how to set up IP multicast and IP unicast, see the discussion on configuring coherence in *Oracle Communications Service Controller System Administrator's Guide*.

About Setting the Service Availability Mode

The components running in the processing tier are stateful; that is, they maintain session information. They retrieve and store session state using in-memory storage to maintain service availability.

When creating a domain, whether you create a unified or a multi-domain, you can select the desired service availability mode:

- **Availability:** Service availability is maintained and continuity is turned off.
- **Continuity:** Service availability is maintained and continuity is turned on.

Important: Turning service continuity off improves performance significantly. However, calls on a failed managed server are not saved.

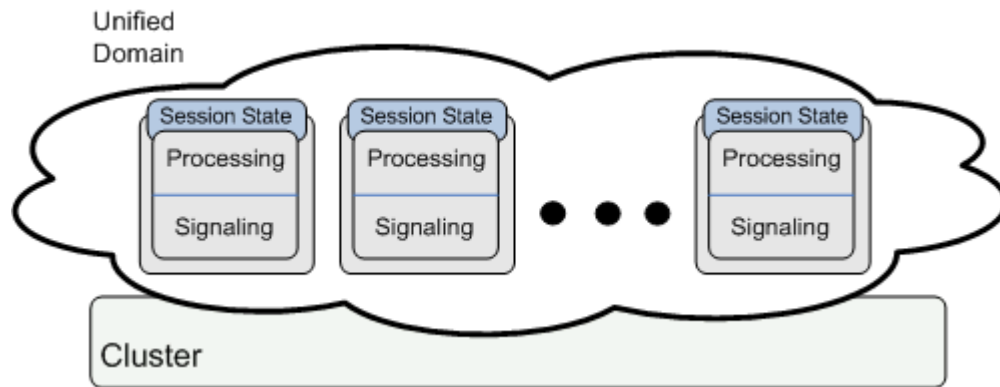
Service availability ensures that if a managed server fails, other managed servers in that domain will handle all new incoming traffic. Service continuity ensures that if a managed server fails, functioning servers continue to retrieve and process all messages, including those stored in the failed server.

About Production Environments

For best performance in production environments, Oracle recommends using a unified domain topology with one domain in which servers perform the role of both the processing and signaling tiers as shown in [Figure 1-4](#).

In this type of deployment, managed servers maintain state information but the state information is not distributed across the servers. Existing calls on a failed server will be dropped. However, another server in the domain will implement service availability and handle new incoming calls.

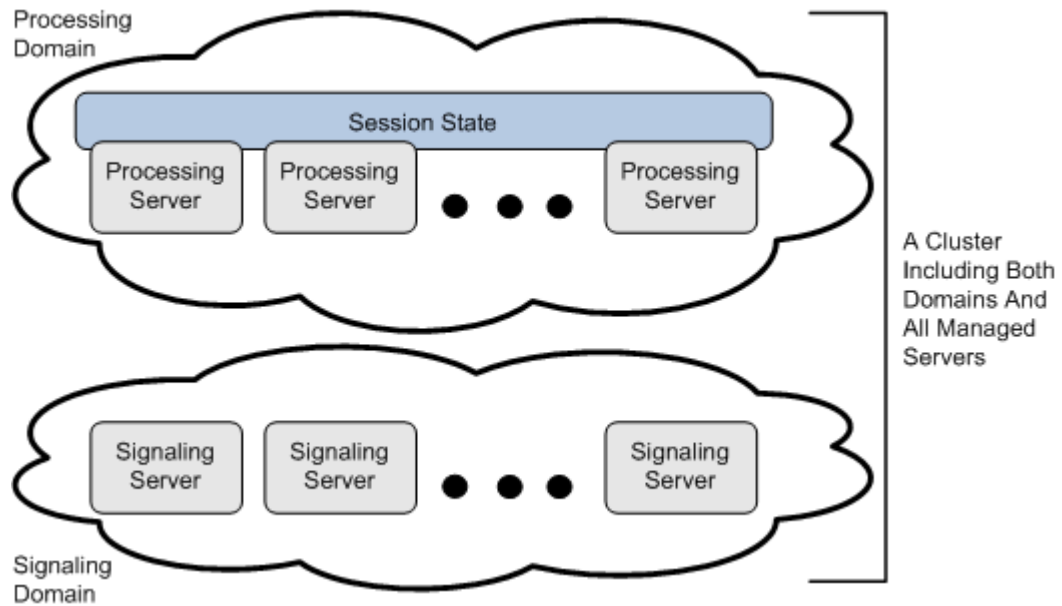
Figure 1–4 A Production Environment Unified Domain



An alternative is to use a multi-domain topology (described in "[Multi-Domain Topology](#)") that implements service continuity, as shown in [Figure 1–5](#).

For complex Service Controller deployments, you can optionally create multiple signaling domains (each connecting to a different network), and multiple processing domains (each supporting a different processing functionality).

Figure 1–5 A Production Multi-Domain Deployment



Note: In a production environment, each domain should always include at least two servers for server failover availability.

About Test and Evaluation Systems

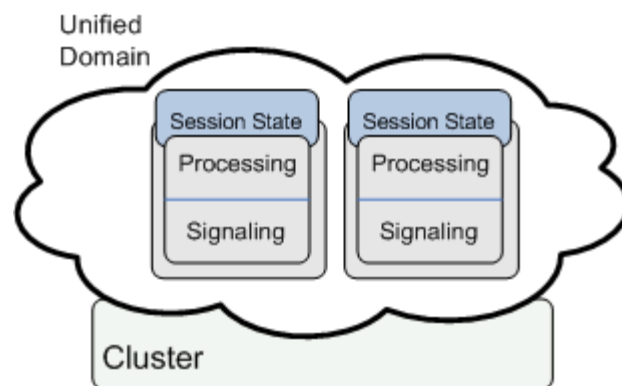
You can use a test and evaluation system to familiarize yourself with Service Controller, for integrations, for proof of concept labs, and for training. A test and evaluation system does not have the same requirements that a production system has for performance, availability, service continuity, and tier redundancy.

A test and evaluation system can be deployed as a unified domain with a single managed server, or two managed servers at most, with each server performing the role of both signaling tier and processing tier.

WARNING: Never convert a test and evaluation system to a production system. A production system should be created independently with the highest degree of care.

Figure 1–6 shows a test and evaluation deployment with one domain. Each server performs the role of both the processing and signaling tiers. In this figure, service continuity is turned off. However, you can also deploy a system with service continuity turned on.

Figure 1–6 Test and Evaluation Deployment



Scaling the Deployment

Scalability is the ability of the Service Controller to handle increasing numbers of requests without adversely affecting performance. Scaling options differ according to whether you are scaling the processing tier or the signaling tier.

Scaling the Processing Tier

The processing tier of a Service Broker deployment includes two or more processing managed servers that together provide high availability and load balancing. The Service Broker processing tier can increase its throughput by adding additional processing servers to the processing domain.

You can add new processing servers by using these scaling options:

- Vertical scaling: You can add to a single physical computer on its processing domain more processing servers.
- Horizontal scaling: You can add a physical computer and deploy processing servers to the same processing domain. The additional processing servers will balance the workload with the processing servers on the other physical computer.

Scaling the Signaling Tier

The signaling tier can also be scaled both vertically and horizontally. You can add signaling servers to the signaling domain on a single computer or by adding new

server host computers. However, there are a few additional items to consider when implementing connectivity to SS7 networks:

- When implementing SIGTRAN-based SS7 connectivity, the signaling tier can include up to sixteen servers whether the servers are administered in a unified domain or in a separate signaling domain.

Note: Note that in a unified domain, your processing tier is limited to the same number of servers.

To maximize scalability, Oracle recommends that you deploy a separate signaling domain to connect to the SS7 network. In so doing, connectivity to other signaling networks is not affected and servers can be added to the other signaling domains as needed.

System Requirements

This chapter describes the hardware and software requirements for Oracle Communications Service Controller.

Hardware Requirements

This section describes the hardware requirements for Service Controller.

Managed Servers

You can deploy managed servers on x86 64-bit CPU architectures.

Each server should have redundant network interface cards.

The following are recommendations for each physical server in a production system. These are guidelines only:

- Dual quad core processors with 16 GB of RAM
- 2 GB of free disk space for the installation

The following requirements apply to specific managed server tiers:

- Processing servers:

Extra disk space is required for processing servers if you store Service Data Records (SDRs) in files on the server. In general, 10 GB is sufficient for handling SDRs. However, the actual disk space you need depends on the server's capacity, the amount of traffic, and how many SDR files are stored on the server.

See information about configuring SDR logging in *Service Controller System Administrator's Guide*.

Administration Server

The Administration Server can be installed on the same physical server as the managed servers; however, Oracle recommends that you install it on its own physical server.

You can deploy the Administration Server on x86 64-bit CPU architectures:

The following are recommendations for each physical computer that hosts an Administration Server. These are guidelines only:

- Quad core processors with 8 GB of RAM
- 300 MB of free disk space for the installation

Software Requirements

The following list describes the software requirements for Service Controller:

- Network Time Protocol (NTP) client

Service Controller requires that all servers accurately synchronize their system clocks to a common time source, to within one or two milliseconds. Oracle recommends using an NTP client or daemon and synchronizing all Service Controller servers to a common NTP server.

Caution: Synchronizing system clocks to within one or two milliseconds is critical for the system to function properly. If you do not synchronize server clocks, the system may display unexpected behavior.

- Service Controller relies upon a shared library of the underlying operating system to provide Stream Control Transmission Protocol (SCTP) support.

If you are implementing a solution that uses Diameter over SCTP, you need to ensure that the operating system includes the following library:

libsctp.so.1

If needed, install the shared library on the operating system of each managed server host machine.

- Oracle Universal Installer imposes additional system requirements and library dependencies. For example, it requires 500 MB of swap space to be available on the installation target system.

For more information about Oracle Universal Installer, see *Oracle Universal Installer and OPatch User's Guide*.

Supported Operating Systems and JDKs

Service Controller can be installed on any of these 64-bit Unix/Linux operating systems:

- Oracle Linux 7
- Red Hat Enterprise Linux 7

Service Controller supports Oracle JDK 64-bit version 1.8 u 162 or later.

For information on other third-party products and their version numbers that are integrated with Service Controller, see *Service Controller Security Guide*.

Supported Web Browsers

The Administration Console can be displayed using any of the following Web browsers:

- Mozilla Firefox
- Google Chrome

Pre-Installation Tasks

This chapter describes the tasks you need to perform before installing Oracle Communications Service Controller.

Downloading the Service Controller Software

You can download the Service Controller software from the Oracle software delivery Web site:

<http://edelivery.oracle.com>

To download the Service Controller software:

1. In a Web browser, navigate to the Oracle software delivery Web site and follow the instructions.
2. When prompted to select a product pack, select **Oracle Communications Applications** and select the platform for your system configuration.
See "[System Requirements](#)" for a description of supported configurations.
3. Select **Oracle Communications Service Controller** media pack and click **Continue**.
4. Follow the instructions to complete the download of the Service Controller media pack.
5. When the download completes, extract the contents of the media pack.

The Service Controller installer is bundled in a Java archive (JAR) file. When you run the JAR file, the Service Controller installer is automatically launched.

Setting Socket Buffer Sizes

To help minimize packet loss, the operating system socket buffers need to be large enough to handle the garbage collection of incoming network traffic.

Service Controller uses Oracle Coherence as its in-memory data grid caching manager. By default, Coherence attempts to allocate a socket buffer of 2 MB. If your operating system is not configured to allow buffers this large, Coherence will utilize smaller buffers. Most versions of Unix have a very low default buffer limit, which you should increase to at least 2 MB.

Coherence will display a warning similar to the following if the buffer size is not set to at least 2 MB:

```
UnicastUdpSocket failed to set receive buffer size to 1428 packets  
(2096304 bytes); actual size is 89 packets (131071 bytes). Consult your OS
```

documentation regarding increasing the maximum socket buffer size. Proceeding with the actual value may cause sub-optimal performance.

It is safe to operate with the smaller socket buffer size, but Oracle recommends that you increase it to 2 MB.

To change the socket buffer size to 2 MB:

1. Log in as root.
2. Open a command shell, if necessary.
3. Run the following commands:
 - `/sbin/sysctl -w net.core.rmem_max=2096304`
 - `/sbin/sysctl -w net.core.wmem_max=2096304`

The first command sets the read memory size, and the second sets the write memory size.

4. (Linux only) Add the Linux **rmem_max** and **wmem_max** name-value pairs as shown in the previous step to the `/etc/sysctl.conf` file to make the configuration change permanent.

Installing the Software

Oracle Communications Service Controller uses Oracle Universal Installer. This chapter describes how to use the Oracle Universal Installer to install Service Controller with either the graphical installation wizard or in silent mode.

About the Service Controller Installer

The installation procedures provide options to install all Service Controller products and components.

You must run the installer as a non-root user. The Service Controller software is installed to the Oracle home directory. See ["Choosing an Oracle Home"](#) for more information.

For more information about using Oracle Universal Installer, see *Oracle Universal Installer and OPatch User's Guide*.

Choosing an Oracle Home

An Oracle home (referred to as *Oracle_home* in all Service Controller documentation) is the parent directory on a host computer into which the Service Controller software is installed.

The *Oracle_home* location is defined in an environment variable.

Each *Oracle_home* has a name and a directory path associated with it that you set during installation:

- The name identifies the program group associated with a particular home and the Oracle services installed on this home. The Oracle home name must be between 1 to 128 characters long, and can include only alphanumeric characters and underscores.
- The path is the full path to an *Oracle_home* directory where products are installed.

If you run the installer for the first time on a host, Oracle recommends that you designate an *Oracle_home* location that is either an empty directory or a non-existing directory for the installer to create.

Oracle Universal Installer supports the installation of multiple *Oracle_home* directories on the same host. However, if you run the installer multiple times, Oracle recommends that you choose the same *Oracle_home* directory for the additional installations to prevent installing multiple versions of the same administrative software.

An *Oracle_home* inventory or local inventory directory contains only information relevant to a particular *Oracle_home*. Each *Oracle_home* directory contains an **inventory** subdirectory. See ["Choosing an Oracle Central Inventory"](#) for more information.

For information on securing the *Oracle_home* directory, see *Service Controller Security Guide*.

Choosing an Oracle Central Inventory

The Oracle central inventory directory contains installation logs corresponding to all installations performed on a particular host. The installation logs for an installation are identified by the timestamp in the log file names.

When you install the first Oracle product on a host, the installer prompts you to specify the location of the central inventory directory. On Linux, this is *Oracle_home/oraInst.loc*.

Choosing an Installation Mode

You can run the Service Controller installer using a graphical installation wizard, or in silent mode using a response file.

In graphical mode, you step through the installation GUI screens, providing information in the dialog boxes when prompted. This method is useful when performing a small number of installations.

In silent mode, your installation choices are determined beforehand in a response file, so you can automate the process of installing multiple servers on multiple hosts. Silent mode is also useful for installing on a host that cannot display the graphical installer.

Installing Service Controller Products in Graphical Mode

To install the Service Controller family of products using Oracle Universal Installer in graphical mode:

1. In a command shell, run the installation JAR file:


```
java -jar ocsc_generic.jar
```
2. The installer displays a series of screens that prompt you for information that is described in [Table 4-1](#). Follow the instructions in the table.

Table 4-1 *Installer Screens in Graphical Mode*

In this screen	Perform the following action
Installation Inventory Setup	Specify the full path of the Oracle inventory directory. Choose the operating system group that has write permissions to the inventory directory.
Welcome	Click Next to proceed with the installation. You may cancel the installation at any time by clicking Cancel .
Auto Updates	Search for updates to existing products. This is an optional step.
Installation Location	Specify the <i>Oracle_Home</i> directory. <i>Oracle_Home</i> is the parent directory to where Service Controller products and components are installed. See " Choosing an Oracle Home " for details about the <i>Oracle_home</i> directory. <i>Oracle_Home</i> may already exist, if you have already installed other Oracle products. In that case, browse to the directory. Make sure that the <i>Oracle_home</i> directory has read and write privileges. If you already have a home directory on your system, you can select that directory (recommended) or create a new home directory.

Table 4–1 (Cont.) Installer Screens in Graphical Mode

In this screen	Perform the following action
Installation Type	Displays all the components that are installed as part of Service Controller. You cannot deselect or change the components. The following products and components are installed: <ul style="list-style-type: none"> ■ Service Controller Admin Server ■ Service Controller Managed Server ■ Service Controller Common Tools ■ Service Controller Others
Prerequisite Checks	Checks that you completed all the preinstallation steps properly. For example, it checks that you deployed the correct JDK to the correct location. If it encounters errors, click the View Log button for more information.
Installation Summary	Displays a summary of the installation, including general information about the installation type, directories, components to be installed, and the total size of all components to be installed. Click Install to run the installer. Click Back if you need to make changes.
Installation Progress	Displays progress of components being installed and scripts being run.
Installation Complete	This screen describes the outcome of the installation. Select Automatically Launch the Configuration Wizard to launch the domain configuration wizard when you close the installation wizard. See " Setting Up a Domain " for more information.

When the installation is completed, the directory structure described in "[Directory Structure After Installation](#)" is created on your server.

Installing Service Controller Products in Silent Mode

This section explains how to install the Service Controller family of products using Oracle Universal Installer in silent mode.

Before you run the installer, specify the installation parameters you want to use, such as the *Oracle_home* directory path and the products and components you want to install, by editing parameters in the **silentInstall.properties** file.

To learn about the parameters, see the descriptions in the "[Installation Parameters in the silentInstall.properties File](#)" section.

Note: The installer validates format and platform compatibility where applicable. However, it will not check spelling so be sure to enter values into the properties file carefully.

If you omit any required value in the properties file, the installer will prompt you to enter the value at the command prompt.

To install Service Controller in silent mode:

1. In a command shell, navigate to the following directory:
installer_directory/operating_system/Disk1/install/
2. Open the **silentInstall.properties** file for editing, and modify the installation parameters as required. [Table 4–2](#) lists and describes the installation parameters.
3. Save and close the file.

- As a non-root user, enter the following command to start the installer:

```
./silentInstaller.sh
```

The installation progress is output to the console.

- If this is the first Oracle product installation on this machine, when prompted, follow the on-screen instructions to run the **oraInstRoot.sh** script located in your `/oraInventory` directory. You must run the script as the root user.

The directory structure described in "[Directory Structure After Installation](#)" is created on your server.

Installation Parameters in the silentInstall.properties File

[Table 4–2](#) lists the installation parameters in **silentInstall.properties** file that the silent installer uses.

Table 4–2 Parameters in the silentInstall.properties File

Parameter	Description
JAVA_HOME	<p>The path to the directory of your Java JDK. If you are installing a JDK with this property file, it is installed in this directory. If you are using your own pre-installed JDK, then reference its path here.</p> <p>The path should be the same as the JAVA_HOME environment variable for your Java installation.</p> <p>No default value is specified.</p> <p>Example:</p> <pre>JAVA_HOME=/usr/local/java/jdk</pre>
ORACLE_HOME_NAME	<p>The name of the installation directory.</p> <p>Mandatory</p> <p>Example:</p> <pre>ORACLE_HOME_NAME=ocsb6.2</pre>
ORACLE_HOME	<p>The full path to your installation directory.</p> <p>Mandatory</p> <p>This path is from <code>/</code> to and including the parent directory of ORACLE_HOME_NAME. If you have previously installed an Oracle product using Oracle Universal Installer, you already have an <i>Oracle_home</i> defined. If you want to reuse the existing <i>Oracle_home</i>, enter its name here.</p> <p>Example:</p> <pre>ORACLE_HOME=/usr/local/oracle_home</pre>

Table 4–2 (Cont.) Parameters in the silentInstall.properties File

Parameter	Description
DEPENDENCY_LIST	<p>Specifies the Service Controller products to install.</p> <p>Mandatory</p> <p>Components to install using the syntax: <code>{"component:version","component:version"}</code></p> <p>Where <i>component</i> is the Service Controller product to install and <i>version</i> is its version. The <i>component</i> may be one of the following:</p> <ul style="list-style-type: none"> ■ service_controller.samples samples for the Service Controller ■ service_controller for the Service Controller ■ ocsb_admin_server for the Administration Server ■ ocsb_managed_server for a managed server ■ existingjdk to specify the location of your own JDK <p>Default value: <code>DEPENDENCY_LIST={"ocsb_admin_server:6.2.0.0.0","ocsb_managed_server:6.2.0.0.0"}</code></p> <p>Example <code>DEPENDENCY_LIST={"ocsb_admin_server:6.2.0.0.0","ocsb_managed_server:6.2.0.0.0"}</code></p>
DECLINE_SECURITY_UPDATES	<p>Indicates whether you want to receive security updates related to Oracle products. Boolean.</p> <p>Mandatory</p> <p>Enter true to decline security updates, or false to receive security updates.</p> <p>No default value is specified.</p> <p>Example: <code>DECLINE_SECURITY_UPDATES=false</code></p>

Table 4–2 (Cont.) Parameters in the silentInstall.properties File

Parameter	Description
SECURITY_UPDATES_VIA_MYORACLESUPPORT	<p>Indicates whether you want to receive security updates through your Oracle My Oracle Support account. Boolean.</p> <p>Mandatory</p> <p>Enter true to receive security updates using My Oracle Support, or false to decline security updates.</p> <p>If you have not registered with Oracle Support, go to the following My Oracle Support Web site and register to obtain a My Oracle Support account: https://support.oracle.com/CSP/ui/flash.html</p> <p>No default value is specified.</p> <p>Example: SECURITY_UPDATES_VIA_MYORACLESUPPORT=true</p>
MYORACLESUPPORT_USERNAME	<p>Specifies your My Oracle Support email address or other email address where you wish to be notified of updates.</p> <p>Mandatory if DECLINE_SECURITY_UPDATES is false or SECURITY_UPDATES_VIA_MYORACLESUPPORT is true.</p> <p>No default value is specified.</p> <p>Example: MYORACLESUPPORT_USERNAME=user@example.com</p>
MYORACLESUPPORT_PASSWORD	<p>Specifies the password for your Oracle support account.</p> <p>Mandatory if DECLINE_SECURITY_UPDATES is false or SECURITY_UPDATES_VIA_MYORACLESUPPORT is true.</p> <p>No default value is specified.</p> <p>Example: MYORACLESUPPORT_PASSWORD=password</p>

Directory Structure After Installation

Figure 4–1 illustrates the high-level directory structure that is created during installation.

Figure 4–1 Installed Directory Hierarchy

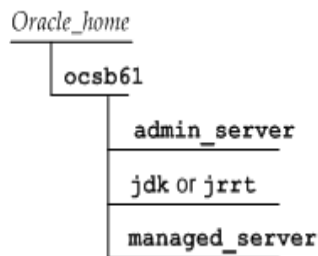


Table 4–3 describes each directory and its contents.

Table 4–3 Installed Directories and Content Descriptions

Directory	Description
<i>Oracle_home</i> /ocsb62	<p>Base directory for your installation.</p> <p><i>Oracle_home</i> is specified during the installation.</p>

Table 4–3 (Cont.) Installed Directories and Content Descriptions

Directory	Description
<i>Oracle_home/ocsb62/admin_server</i>	Contains binaries for the Administration Server, and the start scripts and modules related to the Administration Console.
<i>Oracle_home/ocsb62/managed_server</i>	Contains binaries for the processing server or signaling server, and the start scripts and modules related to the server.
<i>Oracle_home/ocsb62/jdk</i>	Where you install the Java (with the HotSpot JVM) JDK.

Deinstalling Service Controller

It is not possible to deinstall Service Controller.

Setting Up a Domain

This chapter describes how to set up an Oracle Communications Service Controller domain. Before you perform the procedures described in this chapter, you must install Service Controller.

About Domain Setup

Whether a domain is a signaling domain, a processing domain, or a unified domain, setting up a domain requires the following steps:

1. [Understanding the Domain Configuration Directory](#)
2. [Installing the Administration Server and Managed Servers](#)
3. [Securing Domains](#)
4. [Creating a Domain in Interactive Mode](#)
5. [Creating a Domain in Silent Mode](#)
6. [Starting the Domain Web Server](#)
7. [Configuring Security for the Administration Server](#)
8. [Starting The Administration Console](#)
9. [Adding a Managed Server to The Domain](#)
10. [Starting the Managed Server](#)

Understanding the Domain Configuration Directory

Configuration data and software bundles are located in the domain configuration directory. The domain configuration directory is referred to as the *domain_home*.

The domain configuration directory is created when you run the domain creation script to create the domain. You define the method that servers use to access the directory when you create the domain.

The Administration Server and all managed servers in the domain require access to the domain configuration directory.

The Administration Server requires read and write access to the domain configurations. The managed servers require read-only access to the domain configuration.

The Administration Server and managed servers can access the domain configuration directory by using either a hosted or a nonhosted domain. You select either method when you run the domain creation script:

- **Hosted domain:** Uses a Web server to provide managed servers with HTTP or HTTPS access to the domain configuration directory. See "[Starting the Domain Web Server](#)" for more information.

If you use a Web server to provide the domain configuration files, set up the Web server with read access to the domain configuration directory and map the directory to a URL.

Note: The Administration Server requires local file system access to the domain directory because it needs write access to configuration, bundles, credential store and domain lock files.

In a production environment be sure to enable HTTPS, provide redundancy for the Web server, and make backups of the domain configuration directory.

A Web server does not provide any redundancy support by default. A Web server that fails will not impact managed servers that are running, but new managed servers cannot be started until the Web server is back online.

See the discussion on considerations for using Credential Store with hosted domains in the *Service Controller Security Guide* for additional information about working with hosted domains securely.

- **Nonhosted domain:** Connects to the domain configuration directory and the software bundles using a shared file system, not using a Web server.

If you use a shared file system, set up all servers so that they can access the domain configuration directory.

Whether you use a hosted or a nonhosted domain, be certain that all of your restore procedures are rapid to ensure minimal service down time. A recommended solution for high availability is to provide a backup Service Controller installation with a mirrored copy of the domains.

See "[Understanding the Domain-Based Administration Model](#)" for more information about domains.

See *Service Controller Security Guide* for information on securing a domain.

Installing the Administration Server and Managed Servers

You can use the Service Controller installer to install the Administration Server and managed servers on a single computer or on multiple computers. Oracle recommends that you install the Administration Server on its own physical server from where it manages the domain remotely.

Note: Only one Administration Console can be used to manage a domain at a time.

In a unified domain deployment, if you want to install multiple managed server instances on the same physical host, you can do either of the following:

- Run the installer once and then copy the contents of the **managed_server** directory to a new directory for each server you want to add. See "[Creating Multiple Managed Servers by Copying Them](#)" for more information.

- Run the installer multiple times, and other than for the first installation, clear the check box for installing the Administration Server when you select which products and components to install.

Note: Copying the `managed_server` directory is faster than running the installer multiple times.

Installing managed servers on a host and adding them to a domain are separate procedures. After you install the managed server instances, you must use the Administration Console to add the managed servers to a domain. See ["Adding a Managed Server to The Domain"](#) for more information.

Creating Multiple Managed Servers by Copying Them

This section describes how to create multiple servers on a single physical host by copying the managed server directory:

1. Run the installer on the physical host. See ["Installing the Software"](#) for instructions.

The default installed Administration Server directory is:

`Oracle_home/ocsb62/admin_server`

The default installed managed server directory is:

`Oracle_home/ocsb62/managed_server`

2. Create a new managed server directory for each additional managed server.

To simplify your directory structure, you may choose to create the directories alongside the existing server directory in an existing Oracle home. For example:

`Oracle_home/ocsb62/new_server_directory`

3. Copy the contents of the installed server directory to each new server directory that you added.

To create additional servers on another host, repeat this procedure on a different host.

At a later stage, after creating the domain and starting the Administration Server, you must define each managed server in the domain configuration by specifying its name and listening ports.

[Table 5–1](#) illustrates a sample domain with one Administration Server and four managed servers on two physical hosts. The table outlines the directory structure for the example domain.

Tip: To increase availability, you can install the Administration Server on a separate physical host.

Table 5–1 Example Directory Structure

Physical Host	Directory Path	Description
sb_01	<code>oracle_home/ocsb62/admin_server</code>	Administration Server
sb_01	<code>oracle_home/ocsb62/managed_server</code>	Default managed server installation directory Used as the source when creating the directories for the new servers on the sb_01 host.
sb_01	<code>oracle_home/ocsb62/managed_1</code>	First managed server instance directory

Table 5–1 (Cont.) Example Directory Structure

Physical Host	Directory Path	Description
sb_01	<code>oracle_home/ocsb62/managed_2</code>	Second managed server instance directory
sb_02	<code>oracle_home/ocsb62/managed_server</code>	Default managed server installation directory Use as the source when creating the directories for the new servers on the sb_02 host.
sb_02	<code>oracle_home/ocsb62/managed_3</code>	Third managed server instance directory
sb_02	<code>oracle_home/ocsb62/managed_4</code>	Fourth managed server instance directory

Securing Domains

By default, the setting that controls security in the domain is enabled. This setting applies to the connections between the Administration Server and the managed servers in a domain.

Before you run the domain creation script, you must either disable the setting or, if you want to retain security for the domain, set up the client and server certificate keystores on the host system.

For a non-production system intended for test, instructional, or evaluation purposes only, you may choose to disable security between the components, as follows:

1. Open the following document for editing:

`Oracle_home/ocsb62/admin_server/properties/common.properties`

2. Set the `axia.ssl` property to false:

`axia.ssl=false`

For more information on the properties in the file, see *Service Controller System Administrator's Guide*.

3. Save and close the file.

For information on setting up a secure domain, see the discussion on configuring security between Service Controller components in *Service Controller Security Guide*. For information on the property file entries, see the *Service Controller System Administrator's Guide*.

Creating a Domain in Interactive Mode

The domain creation script can operate interactively or in silent mode. The following instructions describe how to use the script interactively. See "[Creating a Domain in Silent Mode](#)" for information about running the script in silent mode.

Important: If you choose to create separate signaling and processing domains, that is a multi-domain topology, you must run the script separately for each domain.

When you run the script, be sure to make the same selections for both the signaling and processing domains. Select the same combination of 1) Available Domains (Service Controller), and the same 2) Service Mode (Continuity, Availability).

To create a domain using the domain creation script in interactive mode:

1. In a command shell, navigate to the `admin_server` directory:

`Oracle_home/ocsb62/admin_server`

2. Run the create domain script, as follows:

```
./script.sh scripts/create_domain_interactive.xml
```

3. Enter the number corresponding to the domain type you want to create.

The options are:

- [0] Unified
- [1] Processing
- [2] Signaling

4. When prompted for the index of the available product domain types, enter the number corresponding to the product domain you want to create.

The options are (example):

- [0] Service Controller Domain

Product domains are displayed only for products you previously installed. You can select one, or any combination, of the product domains.

5. Enter a domain path which can be any valid path to where the domain configuration will be located. Example: **/home/oracle/domains/ocsb-all-products**

This path is referred to as *domain_home* in this document.

6. Do one of the following:

- To create a hosted domain, enter the URL of the Web server that will provide access to the domain configuration: `http://<address>:<port>`
- To create a nonhosted domain, do not enter any value and press **Enter**.

If you select a hosted domain the managed servers will access the domain configuration by connecting to a Web server. If you select a nonhosted domain they will instead need to use a shared file system.

Note that if security is enabled the URL should start with `https://` and the port is defined in the `org.eclipse.equinox.http.jetty.https.port` property in the `hosting.properties` file.

If security is disabled, the port is defined in the `org.eclipse.equinox.http.jetty.http.port` property in the `hosting.properties` file.

7. Enter the IP multicast address for initial Coherence cluster configuration, which is the address on which managed servers and the Administration Server exchange certain types of internal communication within the domain. The address must be within the range 224.0.0.0 to 239.255.255.255.

If you plan to set up additional domains that are part of the same cluster, keep a record of the value entered, and enter the same value when you create the other domains.

8. Enter the port number for initial Coherence cluster configuration on which the servers and the Administration Server listen for multicast messages. The port number must be **1024** or greater.

If you plan to set up additional domains that are part of the same cluster, keep a record of the value entered, and enter the same value when you create the other domains.

9. Enter the time-to-live for multicast messages. The value must be **1** or greater.

10. At the prompt Enable SSL for this domain? (Yes/No) enter **Y** or **N**. In a production environment you should always use SSL. This setting controls whether Service Controller requires a secure connection between deployed domain components.

11. At the prompt for Service Mode enter [1] **Continuity**, or [2] **Availability**. Continuity provides availability and persistent data but can reduce performance. See "[About Setting the Service Availability Mode](#)" for more information.

The following is an example of the domain creation script command and its prompts, with user input in bold:

```
./script.sh scripts/create_domain_interactive.xml
-----
Welcome, please follow the steps below to create a domain.
-----
Available Domain Types:
-----
[0] Unified
[1] Processing
[2] Signaling
Please enter the index of a domain type to create : 0
-----
General Domain Properties:
-----
Please enter the Domain Path : /home/oracle/domains/ocsb-all-products
Please enter the Hosted URL http://<address>:<port> (For non-hosted, press Enter
to skip) :
Please enter the Multicast Address : 239.255.255.255
Please enter the Multicast Port : 1034
Please enter the Multicast Time-To-Live : 2
Enable SSL for this domain? (Yes/No) : y
-----
Additional Domain Properties:
-----
Please select the Service Mode
-----
[0] Continuity
[1] Availability
Please enter the index of the selection : 0
-----
Domain Summary:
Domain Types/Names : Service Controller Unified Domain
Domain Path : /home/oracle/domains/ocsb-all-products
Hosted URL (null if not hosted) : null
Multicast Address : 239.255.255.255
Multicast Port : 1034
Multicast Time-To-Live : 2
Domain SSL Enabled : True
Online Charging Server type : BRM Online Charging Server
Service Mode : Continuity
-----
Proceed with the domain creation? (Yes/No) : y
```

The script creates the domain according to your selections. When the script finishes, the message **Domain created successfully** appears. You can now start the Administration Server and add managed servers to the domain.

Creating a Domain in Silent Mode

You can create a domain in silent mode two different ways:

- Specifying parameters in an XML file
- Entering command-line arguments

Specifying Parameters in an XML File

To create a domain in silent mode, before you run the domain creation script, you must replace all placeholders in the `create_domain_silent.xml` file with valid domain creation values. The XML file is located here: `admin_server/scripts/create_domain_silent.xml`.

When you run the silent install script, the XML file passes domain parameters to the script. If you do not replace any of the placeholders in the XML file with values, when you run the script you will be prompted for the value at the console.

To learn about the parameters, see the descriptions in the section "[Domain Creation Parameters in the create_domain_silent.xml File](#)".

Note: The scripting engine validates format and platform compatibility where applicable. However, it will not perform any spell checking so be sure to enter values into the XML file carefully.

If you omit any required value in the properties file, the installer will prompt you to enter the value at the command prompt.

To create a domain in silent mode using an XML file:

1. In a command shell, navigate to the `admin_server` directory:

```
Oracle_home/ocsb62/admin_server
```

2. Open the `create_domain_silent.xml` file for editing and enter the values you want to use to create the domain. [Table 5-2](#) lists the parameters with descriptions.
3. Save and close the file.
4. Run the create domain script, as follows:

```
./script.sh scripts/create_domain_silent.xml
```

When the script finishes, **Domain created successfully** appears. You can now start the Administration Server and add managed servers to the domain.

Domain Creation Parameters in the create_domain_silent.xml File

[Table 5-2](#) lists the installation parameters in `silentInstall.properties` file that the silent installer applies.

Table 5-2 Parameters in the create_domain_silent.xml File

Parameter	Description
domainTypes	Comma-separated list of domain types to include in the domain. Example: policycontroller-processing,onlinemediationcontroller-processing
domainPath	Absolute file path to the new domain. Example: <code>/home/user/ocsb_home/ocsb62/pc_omc_processing_domain</code>
multicastAddress	Multicast address for initial Coherence cluster configuration. Example: <code>224.0.1.123</code>
multicastPort	Multicast port for initial Coherence cluster configuration. Example: <code>9876</code>
multicastTtl	Multicast TTL for initial Coherence cluster configuration. Example: <code>2</code>

Table 5–2 (Cont.) Parameters in the create_domain_silent.xml File

Parameter	Description
properties	<p>Comma-separated list of properties to allow the specification of default choices and template specific properties. If a property value is not supplied, the default value will be used.</p> <p>If domain.ssl or domain.hosted.url are not included, then the default values will apply (SSL will be enabled and the domain not hosted).</p> <p>Example:</p> <pre>domain.hosted.url=http://localhost:9000/ ,domain.ssl=false,service.mode=availability</pre>

Tip: There are two additional properties that can assist you in preparing a silent configuration:

- "domain.help": If set to true, this property will cause all available options to be printed to the console. It is important that you select the domain types when using this property. The domain will not be created if this property is set to true.
- "domain.summary": If set to true, this property displays a summary of the domain parameters and values before creating the domain. This feature is similar to how the interactive mode summary precedes domain creation.

The next section describes how to create a domain in silent mode by entering arguments at the command-line.

Entering Arguments at the Command Line

To create a domain in silent mode, you must create an AXIA environment variable, assign it parameter-value pairs described in [Table 5–2](#), and then run the silent installation script.

To create a domain in silent mode by entering arguments at a command-line:

1. In a command shell, use the export command to create an AXIA environment variable and set the parameter-value pairs as described in [Table 5–2](#).

Example:

```
export AXIA_OPTS="-Ddomain.types=servicecontroller-processing
-Ddomain.path=/home/user/domains/ocsc-processing
-Ddomain.maddress=225.226.228.229 -Ddomain.mport=1122 -Ddomain.ttl=1
-Ddomain.properties=domain-ocsb,service.mode=availability,domain.ssl=true"
```

2. Run the silent domain script, which picks up the parameter values from the AXIA_OPTS environment variable:

```
./script.sh scripts/create_domain_silent.xml
```

When the script finishes, **Domain created successfully** appears. You can now start the Administration Server and add managed servers to the domain.

Starting the Domain Web Server

If you created a hosted domain, you need to start the domain Web server. If you created a non-hosted domain skip this section.

The domain Web server provides servers with secure HTTPS access to the domain configuration and to the OSGi bundles for the domain. The domain Web server's only functionality is to allow access to the domain configuration.

You specified the domain Web server port when you ran the domain creation script and indicated you wanted to use a hosted domain. The port number is located in the **hosting.properties** file **org.eclipse.equinox.http.jetty.https.port** property.

The domain Web server must be started by a user who has read privileges to the domain configuration directory. See the discussion about configuring security between Service Controller components for more information about setting up user privileges.

To start the domain Web server:

1. Open a command-line shell.

Note: You must be logged in as a user who has read privileges on the file system where the domain configuration resides.

2. Change the directory to:

Oracle_home/ocsb62/admin_server

Oracle_home is the Oracle Home directory you defined when you installed the product.

3. Enter:

.host.sh *domain_home*

Replace *domain_home* with the path to the domain configuration directory.

4. Enter the keystore password.

When the domain Web server is running, the managed servers can access the domain configuration.

Configuring Security for the Administration Server

Starting the Administration Server enables you to start the Administration Console. You can configure the a Service Controller domain from any computer with a browser and network access to the Administration Server.

Note: The instructions below describe setting up default security for the Administration Server to Administration Console connection.

The Administration Console supports a single user. By default the security for this user includes 1) Digest Authentication and an 2) SSL connection between the Administration Console and the Administration Server.

Administration Console password is set by a startup prompt. The first time the Administration Server is started, the user needs to supply a user name and password. These login credentials must be supplied in each Administration Console session.

An alternative security method is described in the Chapter Administering Credential Stores in the *Service Controller Security Guide*.

For more information about Administration Server client authentication, see the Administrator's Reference appendix in the *Service Controller System Administrator's Guide*.

The Administration Console can connect to the Administration Server over (default) Secure HTTP (HTTPS) or HTTP. The following file contains the properties that control Web access, including security for the connection and the Administration Server port:

Oracle_home/ocsb62/admin_server/properties/admin.properties

The default authentication method for the Administration Console is called Digest Authentication (**axia.digest.auth=true**). This setting enables digest authentication which is described in this IEEE RFC, <http://www.ietf.org/rfc/rfc2617.txt>. See the *Service Controller Security Guide* for more information.

By default, the Administration Server is connected on the HTTPS port, as specified by the following setting:

org.eclipse.equinox.http.jetty.https.enabled=true

To use this port, you need to configure the SSL keystore for the Administration Server.

Set this value to **false** to disable the HTTPS. You will also need to enable the HTTP port, as in the following sample portion of the file:

```
axia.basic.auth=false
```

```
org.eclipse.equinox.http.jetty.http.enabled=true  
org.eclipse.equinox.http.jetty.http.port=9000
```

```
org.eclipse.equinox.http.jetty.https.enabled=false  
org.eclipse.equinox.http.jetty.https.port=9000
```

If you are using separate Processing and Signaling domains, you will need to use a separate Administration Server for each domain. Each Administration Server must be assigned a unique port number.

1. Open this file for editing: *Oracle_home/ocsb62/admin_server/properties/admin.properties*
2. Change the port number assigned to the following property:
org.eclipse.equinox.http.jetty.http.port=9000
3. Save the file.

For more information on **admin.properties** and security, see *Service Controller System Administrator's Guide*.

Starting The Administration Console

To start the Administration Console:

1. Open a command-line shell.
2. Change directory to *Oracle_home/ocsb62/admin_server*.
3. Enter the following command to start the Administration Server:

```
./start.sh domain_home
```

where *domain_home* is the path to the domain configuration directory.

4. The first time you start the Administration Server if Digest Authentication (the default setting) is enabled in the **admin.properties** file (**axia.digest.auth=true**) you are prompted to enter a user name and password for subsequent uses of the Administration Console.

The Administration Server enforces password complexity requirements specified in **common.properties**. By default, the password must be at least six characters in length and must contain at least one upper-case letter, lower-case letter, and digit. You can modify the requirements by editing the values of the **axia.console.password.validation** settings in the file.

When the script finishes it displays this message: "Administration Server started".

After starting the Administration Server, you can access the Administration Console in a browser by entering the following URL:

```
https://ipaddress:9000/console
```

See *Service Controller System Administrator's Guide* for more information.

Adding a Managed Server to The Domain

All Managed servers and the Administration Server in a single domain are nodes in a cluster. This is true whether the domain, for example a processing or a signaling domain, exists on a single or on multiple hosts. See "[About Clusters](#)" for more information.

After creating the domain, you need to add to the domain all of the managed servers you installed. You add servers to a domain by using the Administration Console. After you add a managed server to the domain, you will be able to start the managed server, passing it the domain configuration. See "[Starting the Managed Server](#)" for more information.

The following steps provide an overview of adding a managed server to the domain. For more information, see *Service Controller System Administrator's Guide*.

To add a managed server to the domain:

1. Start the Administration Console.
2. Click the **Switch to OFFLINE mode** icon at the top of the page.

Offline mode enables you to make changes to the configuration before adding servers to the domain or when the servers are not running. In online mode, changes are pushed to the managed servers when you commit the changes.

3. In the navigation tree, expand the **OCSB** node.

4. Expand the **Domain Management** node.
5. Click **Servers**.
6. Click the **Add Server** icon.
7. Set the following fields in the **Add Server** dialog as described in [Table 5-3](#).

Table 5-3 Server Configuration Fields

Field	Description
Name	<p>The name of the managed server. This name will be passed in the start command for the server. It must be unique across all domains.</p> <p>In a multi-domain deployment (that is, when using separate processing and signaling domains), server names should follow this format:</p> <ul style="list-style-type: none"> ▪ pn_number for processing managed servers, where <i>number</i> is an integer. For example, pn_1, pn_2, and so on. ▪ ssu_number for signaling managed servers, where <i>number</i> is an integer. For example, ssu_1, ssu_2, and so on. <p>It is possible to use custom server names, but then you must configure a mapping between the custom names to the expected names. See the discussion about mapping custom server names in <i>Service Controller System Administrator's Guide</i> for more information.</p> <p>Format: Alpha-numeric characters. Case-sensitive. No white spaces.</p>
Host	<p>The host name or IP-address of the physical computer where the managed server runs.</p> <p>Format: alpha-numeric. IP-address format or DNS name format.</p>
Admin Port	<p>The IP port used by a managed server to communicate with the Administration Server. The value is assigned automatically and incremented for each managed server.</p> <p>Format: numeric</p> <p>All ports for servers need to be unique for each physical machine.</p>
JMX JRMP port	<p>The port to use for Java Remote Method Protocol (JRMP) invocations to the managed server.</p> <p>Format: numeric</p> <p>All ports for servers need to be unique for each physical machine.</p>
JMX Registry port	<p>The port to use for the MBean Server on the server.</p> <p>Format: numeric</p> <p>All ports for servers need to be unique for each physical machine.</p>

8. Click **OK**.

The new server definition appears in the server list.

Repeat this procedure for each server you installed.

[Table 5-4](#) outlines the corresponding server configurations for the example managed servers whose installation is demonstrated in [Table 5-1](#).

Table 5-4 Example Server Configuration Settings

Server Name	Host	Admin port	JMX Port	JMX Registry
pn_1	sb_01.telco.com	8901	10003	10104
pn_2	sb_01.telco.com	8902	10005	10106

Table 5–4 (Cont.) Example Server Configuration Settings

Server Name	Host	Admin port	JMX Port	JMX Registry
pn_3	sb_02.telco.com	8903	10007	10108
pn_4	sb_02.telco.com	8904	10009	10110

You can now start the server, passing it the identity you configured in the domain, as described in ["Starting the Managed Server"](#).

If you want to make configuration changes while the server is running, you can switch the Administration Server to online editing mode by clicking the **Switch to ONLINE mode** icon. If you intend to continue configuring the domain before starting the server (for example, to configure data store connectivity), you can keep the Administration Server in offline editing mode.

Starting the Managed Server

After you add a managed server to the domain, you can start the managed server. If a domain contains several managed servers, you need to start each managed server individually.

For hosted domains (that is, if you configured the domain files to be exposed by a Web server), you need to start the Web server process first using the **host.sh** script. For more information on hosted domains, see the *Service Controller System Administrator's Guide*.

To start the server:

1. In a command-line shell, change directories to the server directory. By default, the server is installed to the following directory:

```
Oracle_home/ocsb62/managed_server
```

2. Enter:

```
./start.sh server_name domain_home/initial.zip
```

where:

server_name is the name you gave the server when you added the server to the domain.

domain_home is the path to the domain configuration directory where the **initial.zip** file is located. The path may be in the form of a file location or URL. Specify the protocol as follows:

- For a Web-hosted domain, add the protocol prefix **http://** or **https://** to the path, depending on your security settings.
- If your domain configuration is accessed on a shared file system, add the protocol prefix **file://** to the path.

The following is an example of how to start a server, passing it the **pn_1** identity and the configuration bundle located on the local file system:

```
./start.sh pn_1 file:///home/oracle/sc_basic_domain/initial.zip
```

3. If using HTTPS, enter the keystore password when prompted.

[Table 5–5](#) shows the start command and parameters for each server in the example domain introduced in [Table 5–1](#). The start script is located in the server directory.

Note that the name of the server in the domain configuration, as shown in [Table 5-4](#), corresponds to the name given as a parameter when starting the server.

Table 5-5 Example Server Start Commands

Server Directory	Server Start Command for a Non-hosted Domain	Server Start Command for a Hosted Domain
managed_1	<code>start.sh pn_1 file:../domain/initial.zip</code>	<code>start.sh pn_1 http://ipaddress:9001/initial.zip</code>
managed_2	<code>start.sh pn_2 file:../domain/initial.zip</code>	<code>start.sh pn_2 http://ipaddress:9001/initial.zip</code>
managed_3	<code>start.sh pn_3 file:../domain/initial.zip</code>	<code>start.sh pn_3 http://ipaddress:9001/initial.zip</code>
managed_4	<code>start.sh pn_4 file:../domain/initial.zip</code>	<code>start.sh pn_4 http://ipaddress:9001/initial.zip</code>

When you start multiple servers on a single physical host, Oracle recommends that you use a separate command shell for each server.

See *Service Controller System Administrator's Guide* for more information about starting servers.

This chapter provides a concise overview of the essential steps you need to perform after successfully deploying Oracle Communications Service Controller.

After completing the procedures described in the previous chapters, the initial configuration of the Service Controller deployment is complete. You can now configure the functional and operating behavior of the Service Controller, additional features, and advanced topologies.

These are the key steps to perform after installing Service Controller:

- Start the SS7 process if your deployment requires connectivity to the SS7 network
- Perform additional setup tasks required for your particular Service Controller product implementation
- Complete the security configuration for the system
- Configure SNMP

Starting the SS7 Process

If your deployment requires connectivity to an SS7 network, then you need to start the SS7 process on the signaling servers.

After you added a signaling server to a domain, you can start the SS7 process on this server. If a domain contains several servers, you need to start the SS7 process on each server individually.

See the discussion about starting the SS7 process in *Service Controller System Administrator's Guide* for more information.

Performing Additional Setup Tasks for Service Controller Products

You might need to perform additional setup tasks for specific Service Controller product deployments.

See the Service Controller implementation guide applicable to your system for specific setup steps.

When you deploy and configure components in the signaling tier and processing tier as required for the sub-product that you installed and for your specific solution, use *Service Controller Signaling Server Units Configuration Guide* for information about configuring SSUs, and *Service Controller Modules Configuration Guide* for information about deploying and configuring IMs, SMs, and the OE.

Completing Security Configuration for the System

It is recommended that you complete the security configuration for the system, for example, by importing the security certificates that allow secure communications with remote telecommunication network.

See the discussion on configuring security in *Service Controller System Administrator's Guide* and in the *Service Controller Security Guide*.

Configuring SNMP

Configure SNMP settings for sending SNMP traps to external management systems.

See SNMP and other monitoring information in *Service Controller System Administrator's Guide*.