



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

# SIP Feature Server Deployment Guide

SIP Feature Server 812DRAFT

1/24/2024

# Table of Contents

SIP Feature Server Deployment Guide	3
<b>SIP Feature Server 8.1.2 known issues and recommendations</b>	<b>4</b>
<b>Planning and pre-installation</b>	<b>17</b>
Architecture	18
Hardware and software prerequisites	26
Supported Audio Codecs	31
<b>Deploying SIP Feature Server</b>	<b>32</b>
Configure SIP Feature Server applications	34
Deploy a co-located/external Cassandra cluster	45
Configure SIP Feature Server for a co-located/external Cassandra cluster	52
Implement the Feature Server GAX Plugin	53
Implement device management	54
Start SIP Feature Server	57
Configure SIP Server for Feature Server	62
Configure voicemail	64
Configure Regional Voicemail Storage	67
Configure Cassandra Security	70
Add a New Data Center	73
Sample deployments	77
Sample deployments for a co-located/external Cassandra cluster	85
Configuration options	88
<b>Upgrading SIP Feature Server</b>	<b>109</b>
Migrate data from Cassandra database versions	112

# SIP Feature Server Deployment Guide

SIP Feature Server works in conjunction with SIP Server and other Genesys components to provide voicemail, device management, and other services.

You can set up multiple Feature Servers to handle load balancing. You can also achieve high availability through an N+1 configuration.

## Planning and pre-installation

**Plan and prepare** your environment.

---

[Known issues and recommendations](#)

[Architecture](#)

[Hardware and software prerequisites](#)

## Deploying

**Deploy** SIP Feature Server.

---

[Review the configuration options](#)

[View sample deployments](#)

# SIP Feature Server 8.1.2 known issues and recommendations

This document includes SIP Feature Server issues and recommendations applicable to the 8.1.2 release.

---

## No retention limit can't be set when the limit option is configured in days

**Workaround** When using GAX to set retention limits for voicemail profiles, operation can fail when selecting “no limit” option, while retention is measured in days. When retention period specified in seconds then selecting “no limits” option will succeed. (SIPVM-6932)

---

## SIP Feature Server is unable to connect to GAX Plugin

When SIP Feature Server starts in offline mode (no access to internet) and attempts to connect to GAX Plugin, an exception error is displayed. This issue occurs because of incorrectly referenced tiles in template, that is, template DTD (tiles 2) instead of the latest SIPFS tile, that is, tiles 3.

This issue is fixed in SIP Feature Server 8.1.200.83 and later versions.

(SIPVM-6841)

---

## Limitation in usage of hyphen in HOSTNAME

SIP Feature Server doesn't parse the environment variable **HOSTNAME** if it contains ' - ' (hyphen) and it crashed during initialization.

**Workaround:** Genesys recommends not to use ' - ' (hyphen) in **HOSTNAME**. (SIPVM-6873)

---

## Limitation on JDK version with four parts

SIP Feature Server will not start if the JDK version number comprises of four parts, for example, 11.0.14.1.

**Workaround:** Upgrade and/or downgrade to the base JDK version that comprises of three parts, for example, 11.0.18 (latest JDK version with three parts). (SIPVM-6822)

---

## SIP Feature Server doesn't start due to an exception caused by jnr-\*.jar files

SIP Feature Server might not start on some hosts after initial deployment or upgrade from previous version. In such scenarios, the logs contain the following exception error: **Std 09900 Exception java.lang.NoSuchMethodError: jnr.ffi.Platform.getStandardCLibraryName()Ljava/lang/String;**

---

**Workaround:** In the <installation folder>/work directory, locate and remove the **jnr-\*.jar** files, and restart the application. (SIPVM-6714)

---

### Memory leakage in SIP Feature Server

Memory leakage was observed in SIP Feature Server 8.1.202.35 when Prometheus metrics is queried. This issue is fixed in SIP Feature Server 8.1.202.46 and later versions. (SIPVM-6770)

---

### SIP Feature Server will not support the older version of OpenJDK 8

SIP Feature Server will not start with older versions of OpenJDK 8. The use of latest version of OpenJDK 8 is recommended. (SIPVM-6665)

---

### SIP Feature Server will not support Amazon Corretto OpenJDK 11

To start SIP Feature Server, download and configure this version of OpenJDK 11: <https://jdk.java.net/java-se-ri/11>. (SIPVM-6662)

---

### Call Forwarding on No answer doesn't work for external numbers

Call Forwarding on No answer when called using the external number given for an **Agent DN** will not work. This issue exists in SIP Feature Server 8.1.202.00 and later versions. (SIPVM-6468)

---

### Default rate limit value not applied when the values of request-rate-dm, request-rate-vm, and request-rate-gax options are set to alphabet, special characters, Boolean, or negative numbers in SIP Feature Server 8.1.202.40

When you use the 8.1.202.40 version of SIP Feature Server, the default rate limit value is not applied to SIP Feature Server when the options **request-rate-dm**, **request-rate-vm**, and **request-rate-gax** are set to alphabet, special characters, Boolean, or negative numbers. SIP Feature Server can process greater number of DM, VM, GAX requests than their respective default values. When set to a negative value, the rate limit is reached for requests lesser than the default value. (SIPVM-6480)

---

### Difficult to navigate all the mailboxes in the list from the home screen.

When you view accounts that has huge number of mailboxes assigned to it, it is difficult to view all the mailboxes in the list from the home screen.

**Workaround:** For these accounts, Genesys recommends to view their mailboxes from the **User profile** screen or search for a particular mailbox by typing its number in the **Search** field on the **Mailboxes** screen. (SIPVM-6401)

---

### SIP Feature Server's UI has been deprecated from version 8.1.201.83

SIP Feature Server's UI has been deprecated from version **8.1.201.83** dated 09/14/16, and is not supported any further. Therefore, all administrative tasks must be performed using GAX.

---

**Radio button options do not work in SIP Feature Server GAX Plugin UI when you use GAX 9.0.100.XX or later versions**

When you use the 8.1.200.83 version of SIP Feature Server GAX Plugin with GAX version 9.0.100.XX or later versions, you cannot select any radio button option that is available in the SIP Feature Server Plugin User Interface. (SIPVM-6135)

---

**Check box fields will not be visible in SIP Feature Server GAX Plugin UI when you use GAX 9.0.100.XX or later versions**

When you use the 8.1.200.83 version of SIP Feature Server GAX Plugin with GAX version 9.0.100.XX or later versions, the check box fields that are provided for overriding the existing data during Bulk Upload/Bulk Assignment operations will not be visible in the SIP Feature Server Plugin User Interface. (SIPVM-6136)

---

**Voicemail retrieval cannot be performed using PSTN network**

Agents seated outside SBC cannot access the mailbox when they try to access or retrieve the voicemail through PSTN network. The call gets disconnected because PSTN number cannot be resolved by SIP Feature Server. Therefore, retrieving voicemail from remote telephone does not work. (SIPVM-6109)

---

**SIP Feature Server cannot connect to the Configuration Server after switchover**

When the Configuration Server is switched over or when the primary server is down, SIP Feature Server cannot connect to the backup Configuration Server. Therefore, the configuration objects that are created, updated, or deleted at that time are not reflected in SIP Feature Server. This issue occurs for SIP Feature Server versions from 8.1.202.20 to 8.1.202.23. This issue is fixed in SIP Feature Server 8.1.202.24 and later versions. (SIPVM-5988)

---

**Feature Server plugin versions that support GAX version 8.5.260.xx or later**

The following Feature Server plugins support GAX version 8.5.260.xx or later:

- Genesys SIP Feature Server Plugin for GAX 8.1.200.83 or later
  - Genesys SIP Feature Server Device Management GAX Plugin 8.1.200.63 or later
- 

**The fs-nodetool-utility module is not included in the 8.1.202.21 and 8.1.202.22 versions**

The **fs-nodetool-utility** module is not included in the 8.1.202.21 and 8.1.202.22 versions of SIP Feature Server. Therefore, the **nodetool** command cannot be run when secure connection is enabled for Cassandra JMX port. To work around this issue, add **fs-nodetool-utility.jar** from previous versions, manually, and run the required commands.

---

**SIP Feature Server is unable to connect to the configuration server**

SIP Feature Server is unable to connect to the configuration server using auto-upgrade port when TLS

---

support is enabled in the 8.1.202.20 version of SIP Feature Server. To work around this issue, use only the secure port when TLS support is enabled.

---

### **Dialplan will not be supported because of a Jython issue**

The Dialplan TLS connection will not be supported when TLS is enabled in SIP Feature Server due to open issues with Jython library.

(SIPVM-5686)

---

### **LastUpdateID displays null value**

Sometimes, **LastUpdateID** in the Cassandra DB displays null value because of which the connection between SIP Feature Server and Configuration Manager or SIP Server fails. As a workaround, the reimport process must be run to reconfigure **lastUpdateID**. This workaround is applicable only for 8.1.202.17 or earlier versions of SIP Feature Server.

(SIPVM-5690)

---

### **Feature Server will not retrieve voicemail deposited in G729 codecs for 8.1.202.13 or previous versions**

When you try to retrieve the voicemail that is deposited in G729 codecs for 8.1.202.13 or previous versions of Feature Server, the following exception will be thrown:

```
Unsupported audio file
```

(SIPVM-5524)

---

### **Issues in removing obsolete DNs that are assigned to an Agent**

After reimporting Configuration Objects and then while removing obsolete configuration entities, an attempt to remove the obsolete DNs, which are statically assigned to an Agent, from the Cassandra database fails. The issue occurs only during reimport. (SIPVM-4723)

---

### **Issue in upgrading the device firmware using GAX**

In DHCP based provisioning, the AudioCodes phones download the firmware from the URL configured in the 66/160 option. Therefore, the **fs\_url** option in the **[DM]** section does not take effect for Audiocodes firmware upgrade.

(SIPVM-3614, Fix version: Feature Server: 8.1.201.63, DM-GAX plugin 8.1.200.38)

---

### **Null pointer exception during Feature Server History Log synchronization**

When you delete a DN or an Agent Login from the configuration environment when Feature Server is down then a null pointer exception occurs when you restart Feature Server for History Log synchronization. (SIPVM-4563)

---

### Workaround to handle incorrectly updated device in versions 8.1.201.65 or older

When a DN that is assigned to a device is deleted from the configuration environment by using GA/GAX, then the DN remains associated with the device. This issue applies to Feature Server versions 8.1.201.65 or older. (SIPVM-3672, SIPVM-3616, SIPVM-4176, SIPVM-4285)

#### Workaround

Upgrade to the latest Feature Server version, delete the associated device from Feature Server and provision the device again.

---

### Feature Server upgrade from Jetty7.4 to Jetty7.6 has issues

After upgrading SIP FS from FS (running with Jetty 7.4) to FS (running with Jetty 7.6), the FS GAX Plugin displays Access Denied, and attempts to access the FS GAX UI return HTTP error 500.

**Workaround** Remove the old-version jar `org.apache.jasper.glassfish_2.1.0.v201007080150.jsp` from the `<fs-installation-path>/lib` directory and restart Feature Server.

**To fully correct the issue:** Upgrade to the latest FS (running with Jetty v9). The Feature Server from version 8.1.201.86 runs using Jetty v9. (SIPVM-4235)

---

### Upgrade Feature Server from English (ENU) to International (INT) and vice versa in Windows

The setup for Feature Server version 8.1.201.87 for Windows does not recognize the previous Feature Server versions, and as a result, the maintenance update does not occur. Genesys recommends that you install version 8.1.201.88 to correct—or to avoid—this limitation. FS version 8.1.201.88 for Windows recognizes the previous Feature Server versions except 8.1.201.87. Use the following procedure to upgrade to/from version 8.1.201.87.

1. Back up the **cassandra\_home** directory, including the **\etc** subdirectory.
  2. Note the **cassandra\_home** parameter in **launcher.xml** at the current location.  
**Note:** The "current location" is the location of the previously installed FS version that you are replacing.
  3. Run **Setup.exe** from the 8.1.201.88 IP (the new version that you are installing).
    - a. When prompted, enter the Configuration Server host/port and user name/password.
    - b. Select the Feature Server Application object to be upgraded.
    - c. When prompted, enter a new location that is different from the current location.  
**Note:** The "new location" is the location of the latest FS version that you are installing (8.1.201.88).
    - d. Type the Cassandra path as it is specified in the **cassandra\_home** parameter of the current **launcher.xml** (Refer Step 2).
  4. After installing, restore/replace the files located in the **cassandra\_home\etc** directories that you backed up in step 1.
  5. Copy the **launcher.xml** file from its current location to the new location.
  6. Copy the **\resources** directory from its current location to the new location.
  7. Copy the **keystore** file from the current **FS\etc** directory to the new **FS\etc** directory.
-

8. In the new deployment, verify that:
  - a. The Feature Server application object has its new version and working directory.
  - b. **launcher.xml** contains the `cassandra_home` parameter (Refer Step 2).
  - c. The file `\cassandra_home\etc\cassandra.yaml` has the same entries as before. Verify by comparing the new **cassandra.yaml** to the backed-up version (Refer Step 1).
  - d. The **cassandra-topology.properties** file is in the new `\resources` directory—if the `NetworkTopology` strategy is being used.
  - e. Check and adjust the SSL configuration as described on the [Start SIP Server page](#), under the heading **Jetty 9 configuration**.
9. Stop Feature Server from Solution Control Interface (SCI) or Genesys Administrator (GA).
10. Start Feature Server from SCI or GA.

(SIPVM-4259)

---

### Important upgrade step

If you are upgrading from a restricted release of SIP Feature Server 8.1.2 (any version prior to 8.1.200.83), you must manually restore the `vms_host` parameter, as follows:

1. Open `launcher.xml`.
2. In the `vms_host` section, in the line `<format type="string" default="localhost" />`, replace "localhost" with "0.0.0.0" or a specific IP address to restrict Feature Server web application access to that address.

If you are doing a fresh installation or upgrading from 8.1.200.83 or later, you can omit this step.

---

### Synchronization of history fails in non-master Feature Server sites

Synchronization fails following creation / modification / deletion of Switch objects in non-master sites where all the Feature Servers in that site is down, even after all of the site's Feature Servers have been started up again.

**Workaround** Run the Reimport procedure to perform synchronization of those missing switch objects.

(SIPVM-3953)

---

### Permissions Issues Trigger CFGHistoryLogExpired Error

Modifying objects in Configuration Manager when Feature Server is down can trigger a `CFGHistoryLogExpired` error in the Feature Server log, due to permissions issues—even if the number of changes is below the history log max-records limit.

(GSCLOUD-6067)

---

### Agent Login Incorrectly Deleted

---

When an Agent Login is assigned to an Agent/User and the Agent/User is being deleted in CME, Feature Server incorrectly deletes the Agent Login from the Cassandra database.

(SIPVM-3847)

---

### **Agents Not Logged Out**

Agents are not logged out from Audiocodes/Genesys phones if the corresponding line is removed. Recommendation: Do not delete the line before logging out the agent.

(SIPVM-3619)

---

### **Mailbox Counters Show Incorrect Value**

Mailbox counters show an incorrect value while a user performs Voicemail operations.

(SIPVM-3425)

---

### **Omit punctuation from DNs**

Feature Server does *not* ignore punctuation such as commas, brackets, periods in a DN. For example, the dialing instruction in {"instruction": "Dial(2,555)", "priority": 1} is read as "Dial (2)" because of the comma following the 2. Do not include punctuation in a DN when creating or sending dialing instructions to Feature Server.

(SIPVM-1021)

---

### **Install plugin upgrades as new**

The new versions of Genesys SIP Feature Server Plugin for Genesys Administrator Extension and Genesys SIP Feature Server Device Management GAX Plugin must be installed as new—the current upgrade process does not succeed.

#### *Workaround*

Uninstall each plugin first. Then install each new version as a new product.

**Important:** If uninstalling the older plugin versions does not remove the .jar files from the directory <GAX Installation Directory>\plug-ins, then you must remove them manually before installing the latest versions of the plugins.

(SIPVM-3566)

---

If you create a profile with time zone configuration in version 8.1.201.54 or earlier, and then upgrade the Feature Server to version 8.1.201.55 or later, further operations in Device Management may fail after the upgrade.

#### *Workaround:*

---

Reconfigure the time zone in the existing profiles.

(SIPVM-3579)

---

### **Device management: Polycom phones**

- Polycom VVX phones only: When upgrading from Polycom firmware 5.0.0 to 5.0.1, set the **Firmware Upgrade Timeout** to 30.
- When Automatic Call Distribution (ACD) is enabled:
  - Polycom SoundPoint phones with firmware version 4.1.0 or 4.1.1 become unresponsive on agent login.
  - Polycom SoundPoint phones with firmware version 4.0.8 or lower do not display Not Ready reason codes.
  - See **Supported Hard Phones** to determine the best firmware version for your phones.

(SIPVM-3255)

---

### **Multisite: Calls not forwarded between sites**

In a multisite environment, users cannot set call forwarding from one site to another. (SIPVM-3430)

#### *Workaround*

Set the forwarding profile to enable external destinations, and then try to forward to the external version of an internal number: 800-555-7003, for example, rather than 7003.

---

### **Changes to call forwarding settings are sometimes ignored**

If an agent sets call forwarding from their agent desktop application, such as Workspace, that setting immediately synchronizes to Feature Server. However, changes to call forwarding settings on Feature Server made through the GAX-based UI are not synchronized back to SIP Server. SIP Server retains the previous forwarding settings, resulting in unexpected behavior. (SIPVM-3409)

#### *Workaround*

Have agents choose only one location to set and change forwarding: either their agent desktop application, or the Feature Server UI.

---

### **Changing a forwarding profile needs page refresh**

After the selection of a new forwarding profile for a user, the User Properties page might not reflect the change. (SIPVM-3383)

#### *Workaround*

Refresh the browser page to view the proper settings for that profile.

---

### Do not rename switches

Renaming a switch causes Feature Server to treat the switch as a new switch, creating duplicate data in Cassandra. Do not rename any switch.

---

### Device management: Firmware upgrades

- Firmware upgrades initiated for a device may fail if the upgrade timer expires during a call.
- In business continuity deployment, a peer site cannot upgrade the firmware of phones belonging to the original site.
- If Feature Server terminates during a firmware upgrade, and the firmware upgrade completes after Feature Server is running again, the firmware version and upgrade status are not properly updated.

#### Workaround

Resync or restart the device to correctly update the Device Firmware version and upgrade status.

---

### Device management: Audiocodes SBC

Provisioning of multiple lines on a device behind an Audiocodes SBC may not work correctly. (SIPVM-3392)

#### Workaround

Use an Audiocodes SBC version 6.80A or later.

---

### Device management: Bulk assignment

You cannot use bulk upload to overwrite a device by interchanging the DNSs. (SIPVM-3338)

---

### Device management: Synchronization

- A re-enabled device can fail to synchronize for up to an hour, until the periodic request is sent for the configuration file. (SIPVM-3330)

#### Workaround

After enabling a disabled device, an administrator can restart the device manually for immediate effect.

- When an assigned DN is disabled and then deleted in Genesys Administrator, synchronization of the device cannot occur until the device is rebooted.
- 

### Device management: Yealink phones

- You can disable and enable only phones that are updated to firmware 34.72.0.20 or later. (SIPVM-3401)
  - In a Business Continuity environment where one site is inoperative, ACD agents using Yealink phones cannot log in. (SIPVM-3285)
  - When you disable Call Forwarding using the Feature Server device profile **Call Settings**, phone users
-

can still forward calls from Yealink phones.

---

### **Device management: Agent login**

ACD agents cannot effectively log in from multiple devices. The device accepts their credentials, but their only option is to log out. (SIPVM-3267)

---

### **Device management: LDAP**

- Genesys and AudioCodes phones: users cannot use the Number Attribute to search LDAP directories. (SIPVM-3218)
  - Active Directory (LDAP) is not supported for Yealink Model T-20.
  - To use LDAP, Polycom phones with firmware versions below 4.x require an appropriate license.
- 

### **Notifications: Firefox formatting**

Notifications: in the Firefox browser, the email and web message bodies do not retain line breaks. (SIPVM-3001)

---

### **Device management: IVR Provisioning**

High availability for IVR provisioning is not supported. For example, if Feature Server terminates during IVR provisioning, the administrator must re-provision the phone.

---

### **Device management: HTTPS**

AudioCodes phones do not require CA certification configuration for Hypertext Transfer Protocol Secure (HTTPS) support.

---

### **Feature Server and GAX Server synchronization**

You must always keep Feature Servers and GAX Servers synchronized within one second.

---

### **Keep one Feature Server instance running at all times**

To avoid discrepancies between the Cassandra and Configuration Server databases, keep at least one Feature Server with active confSync running at all times.

---

### **Upgrade steps**

If you are running the Feature Server-based dial plan, to upgrade your environment from 8.1.200.88 to 8.1.201.xx you must take the following actions:

1. [Run the Voicemail Enabled migration script.](#)
-

2. A user setting of Unconditional Forwarding in 8.1.200 sets an empty value for Forward All Calls To; all calls automatically go to voicemail. You must add the voicemail access number to Forward All Calls To. (SIPVM-2320)
  3. If the switch-level forwarding options in 8.1.200 were set to Voicemail, after upgrade these values are System (Voicemail). The administrator must set custom values at the switch level for Forwarding On Busy and Forwarding On No Answer, save the values, then reset those values to System (Off). (SIPVM-2315)
- 

### Feature Server GAX Plug-in does not support HTTPS

Feature Server GAX Plug-in does not support HTTPS URLs. (SIPVM-2852)

#### Workaround

Specify only http URLs as values for the **fs\_urls** configuration option.

---

### Multiple data center environment: exception occurs in nodes during startup

During the initial startup of a multiple data center environment, a `TokenRangeOfflineException` occurs, with a log message "[Voicemail] Failed to read mailbox". The cause is a **voicemail-quorum** configuration option value of `true`. (SIPVM-2840)

#### Workaround

When you first start Feature Server in a multiple data center environment, set the **[VoicemailServer]voicemail-quorum** configuration option value to `false`. After startup is complete, reset the value to `true`.

---

### Master Feature Server can wait indefinitely for other nodes to start

After the master Feature Server starts, it sometimes waits indefinitely in the initializing state, waiting for other nodes to be started so it can retrieve all system data about the Cassandra cluster.

#### Workaround

If the master Feature Server is waiting at the initializing state, start the other Feature Server nodes in the Cassandra cluster.

---

### Too many non-functioning Cassandra nodes can lead to inconsistent data

In a multi-node environment, if the option **voicemail-quorum** = `true` and the number of non-functioning Cassandra nodes is greater than or equal to the calculated quorum value (**replication\_factor**/2 + 1, rounded down), Feature Server may return MWIs with incorrect counts, because not enough nodes are available to apply Cassandra Read /Write Consistency policies.

#### Workaround

Do not take multiple Feature Server instances offline at one time.

---

## **Installation on Windows 2012**

Before attempting to install Feature Server under Windows 2012, verify that you have the latest Feature Server 8.1.201.xx installation package (IP) or CD. Open the IP Readme located on your CD or IP. If the Readme does not list Windows 2012 support, then you must obtain the latest Feature Server 8.1.201.xx IP.

---

## **Cassandra cluster outages can cause data synchronization issues**

If the Cassandra cluster is not operational at all times, data synchronization-related issues in Feature Server instances can occur. (SIPVM-2280)

---

## **Forwarding On Busy or No Answer: calls not depositing voicemail to original mailbox**

When a user or DN has Forwarding On Busy enabled (On), callers are unable to deposit voicemail into the mailbox of the user or DN they originally called. When Forwarding On Busy or No Answer is set to EXTERNAL\_PUBLIC, the call does not return to the original mailbox. (SIPVM-2135)

---

## **Non-agents are incorrectly having mailboxes created automatically**

In a standalone deployment, non-agents are incorrectly having mailboxes created automatically during initial import and real-time synchronization of data into Cassandra from Configuration Server. (SIPVM-2134)

---

## **Calls to agents with multiple mailboxes are sent to the first associated mailbox**

When a person is statically associated to two agent logins, each of which has its own mailbox configured, all calls are sent to the first associated Agent Login mailbox. (SIPVM-2124)

---

## **Use IP addresses**

Use IP Address or local host name, not FQDN, to access the Web application. FQDNs can cause unexpected logouts.

---

## **Group Mailbox Administrator privileges**

Only users with the Group Mailbox Administrator role can use the web application or TUI to upload or change greetings and passwords.

---

## **When the network is disabled on the host on which Feature Server is running, Windows may terminate unexpectedly**

Windows may terminate unexpectedly if the host on which Feature Server is running is removed from the network by clicking Disable from the Windows Local Area Connection dialog box. (SIPVM-1333)

*Workaround*

---

Stop Feature Server before disconnecting the network.

---

**Message priority is not being considered during retrieval**

The message priority selected during a call deposit is not taken into consideration during voicemail retrieval. (SIPVM-1248)

---

**TUI: mailboxes are accessible only with mailbox credentials**

In the Telephone User Interface (TUI), User and Group mailboxes can be accessed only with mailbox credentials (mailbox number). DN, agent, and user credentials to access mailboxes is not supported. (SIPVM-859)

# Planning and pre-installation

Before installing and configuring SIP Feature Server, you must plan your environment and install required hardware and software.

In your planning, you must:

- Meet [hardware and software prerequisites](#) such as operating system, hardware, and Genesys and third-party components.
- Review the current [known issues and recommendations](#) for this release.

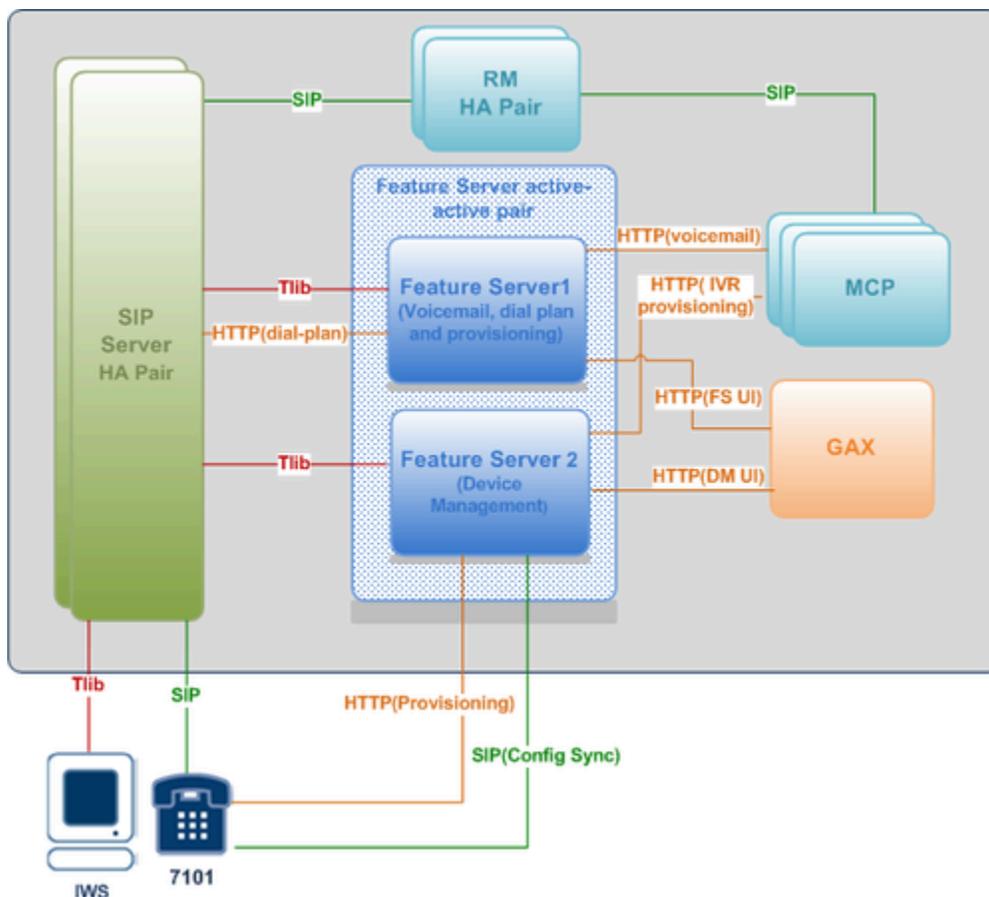
# Architecture

SIP Feature Server provides a scalable, load-balanced architecture that enables you to add Feature Server instances as your needs expand. A Cassandra-based data cluster maintains data consistency and performance across all instances.

## Deployment options

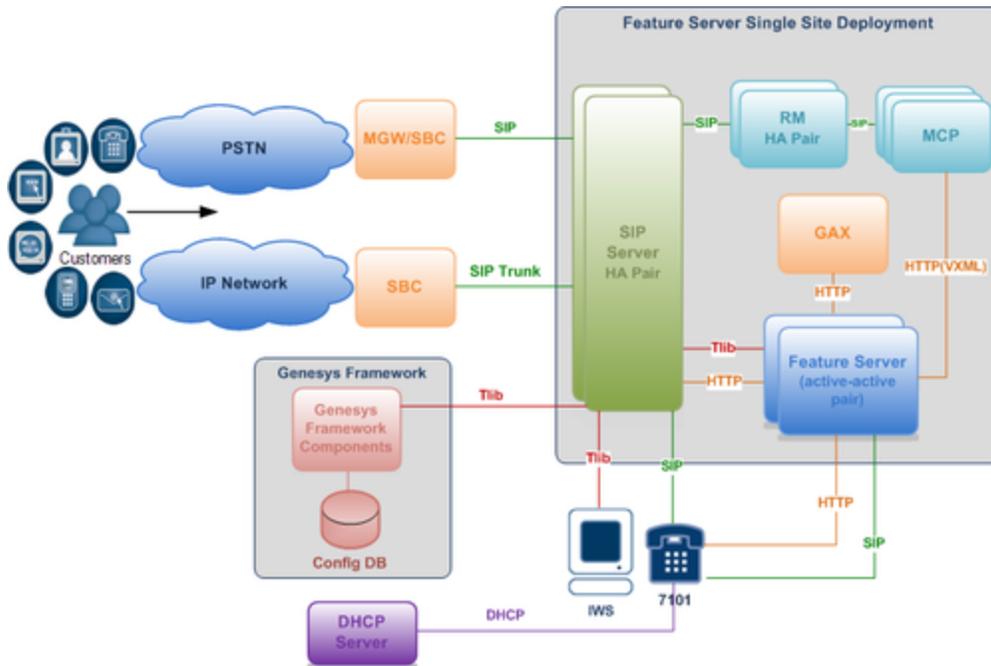
Standard Feature Server deployment includes a High Availability (HA) installation with a SIP Server pair and at least two Feature Servers, one primarily dedicated to voicemail, dial plan, and provisioning, and the other to device management. Multi-site (multi-switch) deployments replicate that basic structure for each switch. Business Continuity adds a further layer of reliability.

You can use [these sample deployments](#) as a model for your chosen deployment option.



Core Feature Server architecture

### Single-site deployment

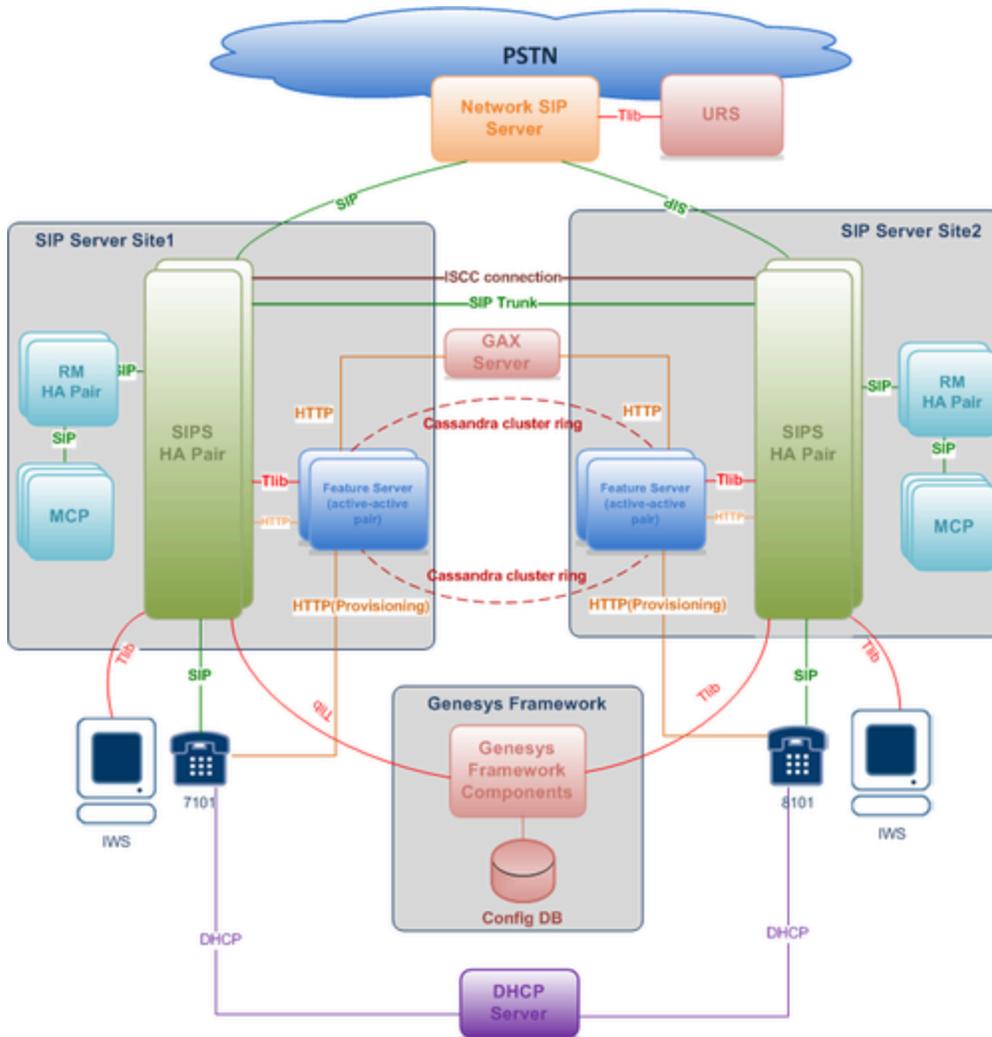


Standard single-site Feature Server architecture

Each Feature Server connects to only one SIP Server, but all Feature Servers use a common Cassandra cluster that sites can share, enabling any user to access voicemail from any site. Further, because all the Feature Server nodes share the same Cassandra database, if one Feature Server is down, the other Feature Servers can operate without loss of functionality.

To learn more about SIP Server HA deployment, see [SIP Server 8.1 High-Availability Deployment Guide](#).

### Multi-site deployment



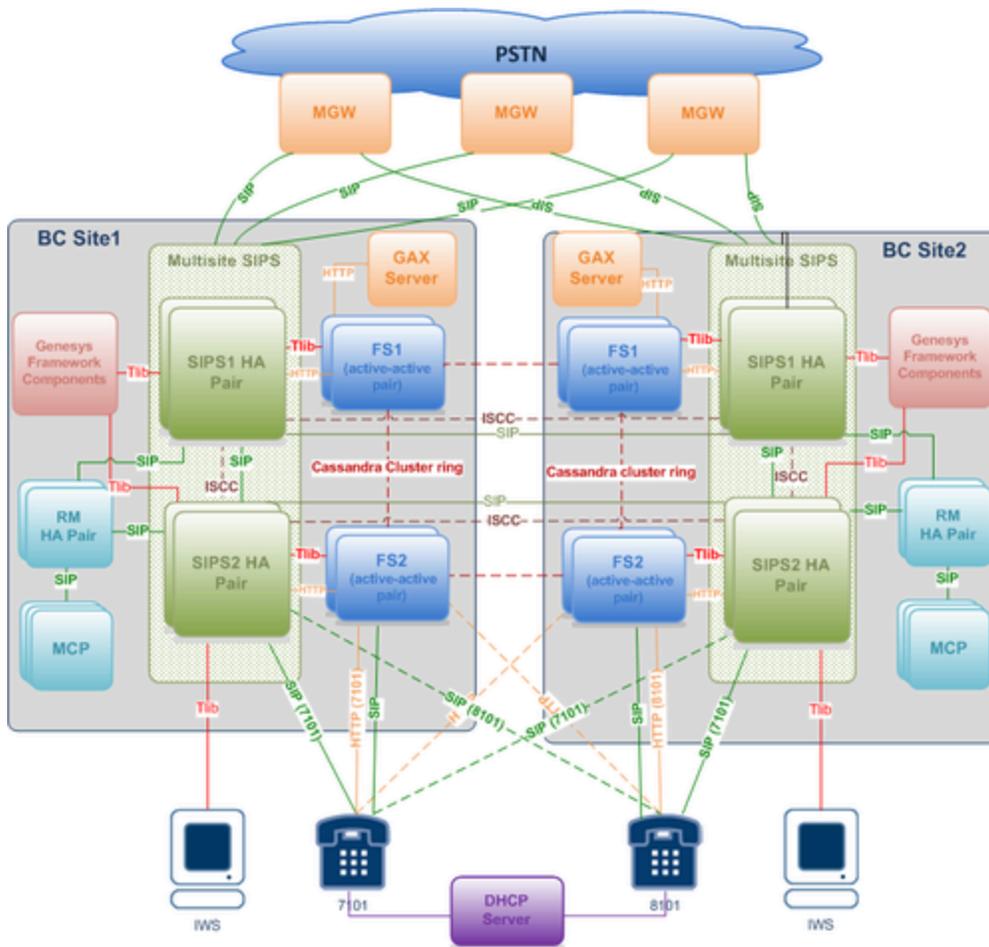
Standard multi-site Feature Server architecture

You can deploy SIP Feature Server across multiple data centers, in an active-active configuration. Feature configuration is the same for both multi-site and single-site deployments.

You can install Feature Server in a multi-tenant framework, but each Feature Server deployment can service only a single Tenant.

### Business Continuity deployment

Business Continuity provides the ability for a group of agents to continue offering critical business functions to customers in the event of a loss of all Genesys components running at a particular site. Specifically, phones can connect and be administered on either site when the other is inoperable.



Feature Server Business Continuity architecture

### Important

Business Continuity deployments do not support IVR provisioning.

Feature Server Business Continuity deployment begins with a **SIP Server Business Continuity deployment**, then adds the standard Feature Server multisite deployment. Each site acts as the backup for the other.

Business Continuity deployment requires you to create at least one device profile for each site. You then specify **Business Continuity settings** for each profile.

### Provisioning and dial plan setup

Initial program configuration occurs primarily in Genesys Administrator (GA). You can create users, DN, and mailboxes only in GA. To set up your dial plan, a highly configurable set of call disposition

---

patterns, you can select either of two methods:

- the Feature Server plug-in for Genesys Administrator Extension (GAX)
- the existing SIP Server methodology, using GA

Administrators can also use the Feature Server GAX plug-in to provision users, devices, voicemail, and call settings.

### Important

A Feature Server's dial plan URL must be configured only on a VOIP Service DN that was created on the Switch controlled by the SIP Server that is connected to that particular Feature Server. Step 2 in [Configure SIP Server for Feature Server](#) describes creating the VOIP Service DN.

## Voicemail

SIP Feature Server combines with Genesys Voice Platform (GVP) and SIP Server to handle voicemail tasks.

The Feature Server GAX plug-in and the Telephone User Interface (TUI) enable users to specify mailbox settings and manage their voicemail.

### Important

A single Feature Server GAX Plugin cannot serve multiple tenants.

## Device management

The Device Management GAX Plug-in provides an administrative interface for provisioning and management of SIP phones, including extension assignment, standard and custom configuration, firmware updates, and IVR provisioning. Feature Server supports phones behind SBCs and firewalls.

## Data management

### Important

Beginning with SIP Feature Server version 8.1.203.XX, the **Embedded Cassandra cluster** mode is being deprecated and the feature will be completely removed in future versions.

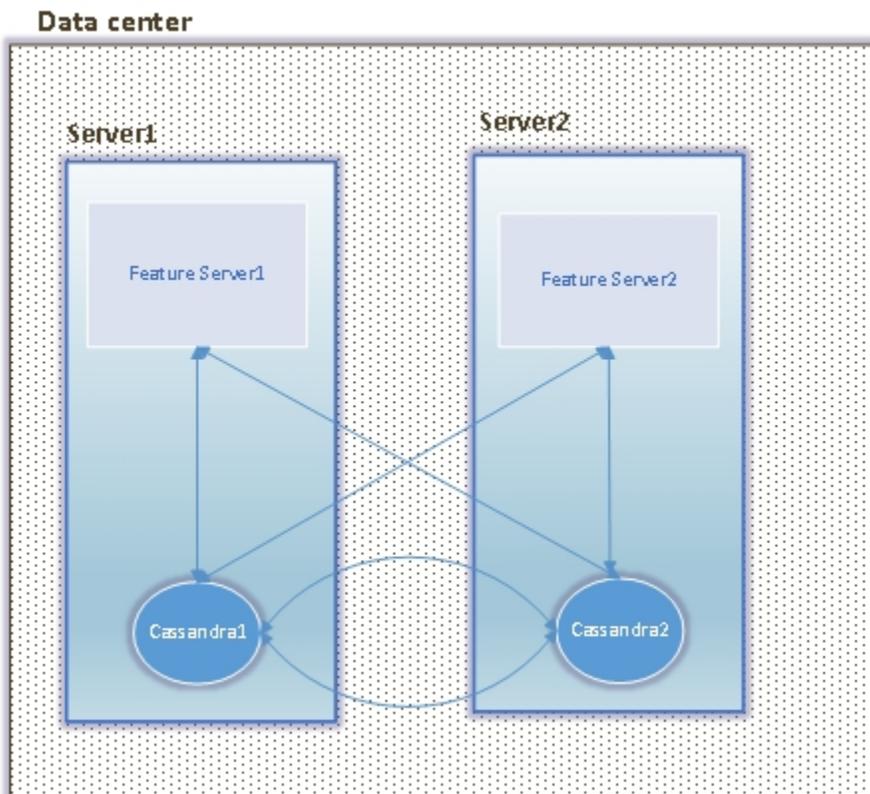
SIP Feature Server uses **Apache Cassandra** data clusters to replicate data across the environment, achieving scalability and high availability. Feature server supports the Co-located/External Cassandra cluster mode.

In co-located/external Cassandra cluster:

1. Cassandra must be installed and configured separately.
2. SIP Feature Server supports Cassandra version 2.2.8 or higher.
3. SIP Feature Server does not control Cassandra monitoring tasks such as start, restart, or stop. This control is performed separately.
4. SIP Feature Server supports secure connection with Cassandra.

### Co-located Cassandra cluster

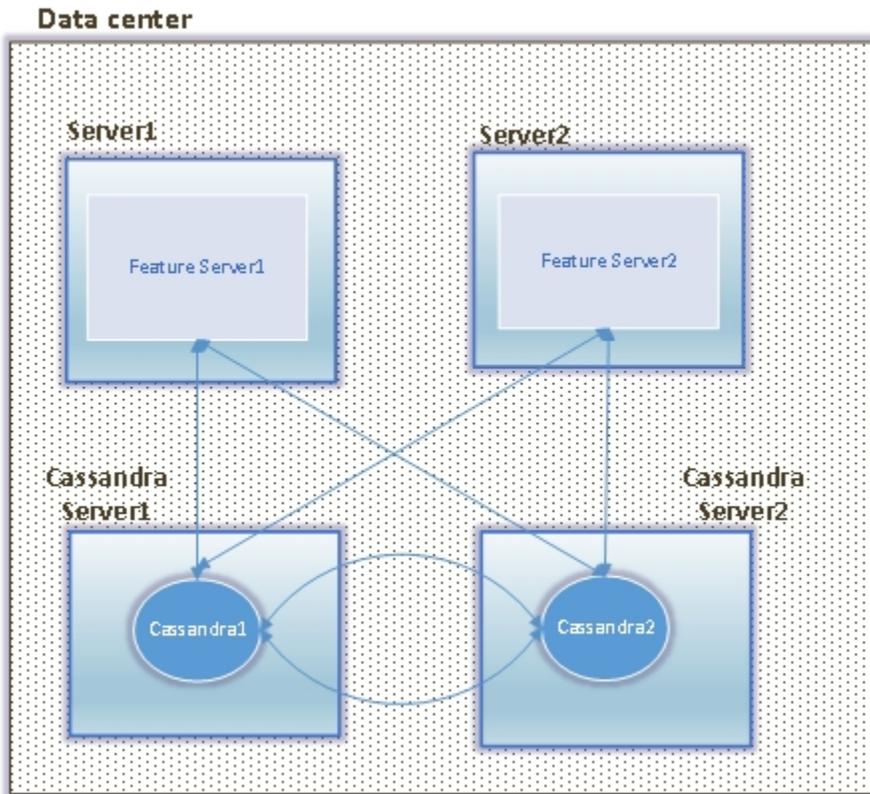
In the Co-located Cassandra cluster mode, Cassandra is deployed separately but in the same host where SIP Feature Server is deployed.



SIP Feature Server with co-located Cassandra cluster

### External Cassandra cluster

In the External Cassandra cluster mode, Cassandra is deployed separately and in a different host.



SIP Feature Server with external Cassandra cluster

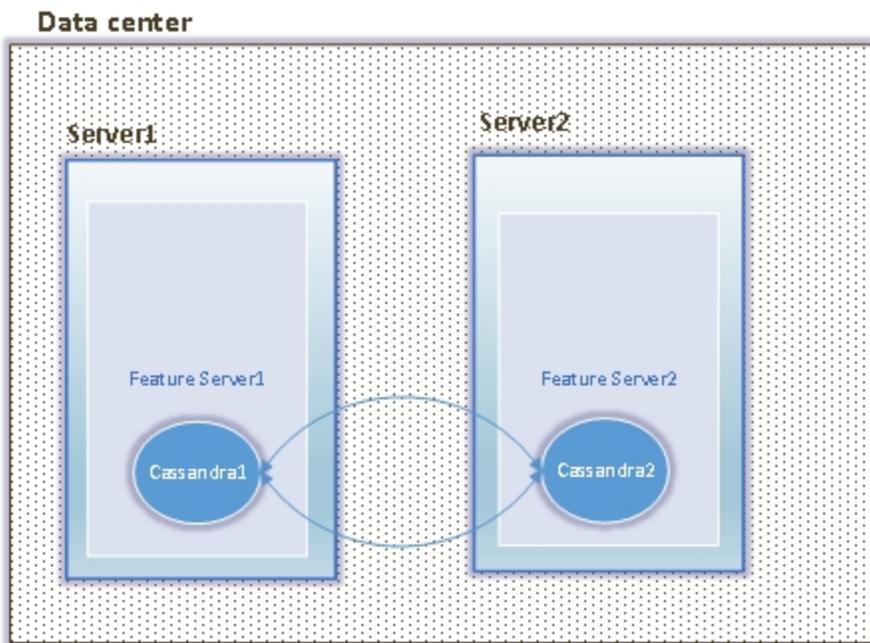
### Legacy Cassandra Cluster Deployments

Legacy Cassandra cluster deployments include Embedded Cassandra cluster which is being deprecated and it will not be supported in future versions. If your deployment model involves embedded Cassandra cluster, Genesys recommends to migrate your data with other Cassandra cluster deployments such as Co-located or External Cassandra clusters.

In the Embedded Cassandra cluster deployment:

1. SIP Feature Server uses its own embedded Cassandra and Cassandra is not required to be installed separately.
2. Cassandra starts when SIP Feature Server is started.
3. SIP Feature Server supports Cassandra monitoring and maintenance.

The SIP Feature Server installer deploys the embedded Cassandra cluster.



SIP Feature Server with embedded Cassandra cluster

# Hardware and software prerequisites

Before you install Feature Server, ensure that your system is equipped with the hardware, software and other related requirements detailed in this section.

## Hardware

The following table lists the hardware requirements:

Category	Prerequisite
Host	Install Feature Server on its own dedicated host, unless you are deploying to a small or lab environment. Do not co-locate any other applications on this host.
Operating system	<p>To view supported operating systems, see the <a href="#">Genesys Supported Operating Environment Reference Guide</a>.</p> <ul style="list-style-type: none"> <li>All Feature Servers in your environment must run on either Linux or Windows. You cannot mix Linux and Windows machines.</li> <li>If you are running RHEL 7/8, you must:                             <ul style="list-style-type: none"> <li>Install all RHEL 7/8 compatibility packages</li> <li>Install libcurl.so.3 (64-bit only) (Feature Server requires libcurl 3.0.0 to start).</li> <li>Create a symbolic link for libcurl.so.3 (64-bit only):                                     <pre>cd /usr/lib64  ln -s /usr/lib64/libcurl.so.4.3.0 ./libcurl.so.3</pre> </li> </ul> </li> </ul>
RAM	4GB of RAM or above available to the Java process.

## Software

The following table lists the software requirements:

Category	Prerequisite
Java Runtime Environment (JRE)	<b>Supported Java Runtime Environments:</b> JRE 7, JRE 8, OpenJDK 8, and OpenJDK 11.

Category	Prerequisite
	<p><b>Java heap size</b> By default, the Java heap size is set to 1GB in <b>launcher.xml</b>. To avoid java heap memory error, increase the Java heap size to 1/4th of the system's RAM size as this is the maximum permissible Java heap size. To increase the Java heap size, update the default value of the <b>jvm_option2</b> option in <b>launcher.xml</b>. Below is an example <b>launcher.xml</b> file for a Feature Server machine that has 16GB RAM with a 4GB java heap size.</p> <pre data-bbox="824 533 1393 835"> &lt;parameter name="jvm_option2" displayName="jvm_option2" mandatory="true" hidden="true" readOnly="true"&gt;   &lt;description&gt;&lt;![CDATA[Cassandra related]]&gt;&lt;/description&gt;   &lt;valid-description&gt;&lt;![CDATA[]]&gt;&lt;/valid- description&gt;   &lt;effective-description/&gt;   &lt;format type="string" default="-Xmx4g" /&gt;   &lt;validation&gt;&lt;/validation&gt; &lt;/parameter&gt; </pre> <p><b>Note:</b> If you are using the <b>launcher_64</b> executable, you will need to update <b>launcher_64.xml</b> instead of <b>launcher.xml</b>.</p>
<p>Genesys requirements</p>	<p>Install and configure:</p> <ul data-bbox="834 995 1446 1759" style="list-style-type: none"> <li>• <b>Genesys Management Framework:</b> See the <a href="#">Genesys Administrator Management Framework Deployment Guide</a> for details.</li> <li>• <b>Genesys Media Server 8.5.x (recommended) or 8.1.5 (minimum) components:</b> Resource Manager and Media Control Platform. See <a href="#">Genesys Media Server 8.5 Deployment Guide</a>.</li> <li>• <b>Genesys Administrator Extension:</b> 8.1.400.59 or higher. See the <a href="#">Genesys Administrator Extension Deployment Guide</a> for details.</li> <li>• <b>Genesys SIP Server</b> <ul data-bbox="878 1436 1446 1759" style="list-style-type: none"> <li>• <b>A SIP Server 8.1.1 (or later) instance for managing agents:</b> ACD functionality requires SIP Server 8.1.101.56 or later. Note: If you want to use an existing premise SIP Server to also process voicemail, you must use SIP Server version 8.1 or higher.</li> <li>• To provision phones with Feature Server version 8.1.201.52 or later requires SIP Server version 8.1.101.75 or later. See the Genesys SIP Server Deployment guide for details.</li> </ul> </li> </ul> <p style="text-align: right;">See the <a href="#">Genesys SIP Server Deployment</a></p>

Category	Prerequisite
	<p><a href="#">Guide</a> for details.</p> <ul style="list-style-type: none"> <li>• <b>All application templates:</b> Use the supplied templates for SIP Feature Server.</li> </ul>

## Device management

You can manage SIP desk phones from Polycom, AudioCodes, Genesys, and Yealink using the [SIP Device Management](#) area of Genesys Administrator Extension (GAX). If you require this feature, then install and configure the following:

- Syslog server (mandatory)
- LDAP server (optional)
- NTP server (optional)

For details on the implementation, see [Implement device management](#).

## Sizing

Feature Server provides three tools to help you size your environment:

- To calculate the database (embedded Cassandra) disk space requirements for voicemail and device management, enter your information in column B of the [SIP Feature Server 8.1.2 Sizing Guide](#). The guide also calculates network disk space needed for device management functionality.
- To calculate sizing requirements that match your performance needs, [use this Feature Server Sizing Tool](#).
- In the case of a co-located/external Cassandra, Feature Server requires a minimum of one Cassandra node per Feature Server pair. However, it is also based on the number of replicas that are required in a data center.
- For help with other sizing requirements, view [Voicemail Performance Test Results](#).

## External Cassandra

If Genesys SIP Feature Server is configured with an external Cassandra cluster, then the Cassandra nodes require the following hardware and software.

Category	Requirements
Operating System	All Cassandra nodes in your environment must run on either Linux or Windows. You cannot mix Linux and Windows machines.

Category	Requirements
	The clocks on all the Cassandra nodes and Feature Server must be synchronized.
RAM	8 GB of RAM or above.
Runtime environment	For Cassandra 2.2.x, Java Runtime Environment (JRE) v1.7 64-bit or higher For Cassandra 3.x, Java Runtime Environment (JRE) v1.8 64-bit or higher
User resource limits	For Linux environment, ensure that the following user resource limits are present and use the <code>ulimit -a</code> command to view the current limits: <ul style="list-style-type: none"> <li>• memlock - unlimited</li> <li>• nofile - 100000</li> <li>• nproc - 32768</li> <li>• as - unlimited</li> </ul>

## Ports

### Important

Do not use the following ports for any other applications that share the host on which you install Feature Server/Cassandra.

## Feature Server Nodes

Feature Server uses the following default ports.

Port	Used for
8080	http
8443	https
8800	Dial plan

## Cassandra Nodes

Cassandra nodes use the following default ports.

---

Port	Used for
7000	Cassandra Storage_port
7001	Cassandra SSL_Storage_port
9160	Cassandra rpc_port
7199 (External Cassandra) or 9192 (Embedded Cassandra)	JMX remote port
9042	CQL port (Cassandra v2.2)

## Supported Audio Codecs

Feature Server supports the same audio codecs that are supported by Genesys Media Server. The list below includes the codecs and the associated file names.

**Note:** The G.722, G.729, AMR, and AMR-WB codecs are disabled by default. You can enable these codecs by using the codecs option in the mpc section. If the default value for this option is retained, the less resource-intensive codecs are used first.

**Supported Audio Codecs and File Names**

Audio codec name	Name in	File name
AMR	amr/8000	<name>.amr
AMR-WB	amr-wb/16000	<name>.amr-wb
G.711 Mu Law	pcmu/8000	<name>_pcmu.<ext> or <name>_mulaw.<ext>
G.711 A Law	pcma/8000	<name>_pcma.<ext> or <name>_alaw.<ext>
G.722	g722	<name>_g722.<ext>
G.726-16	g726-16/8000	<name>_g726-16.<ext>
G.726-24	g726-24/8000	<name>_g726-24.<ext>
G.726-32	g726-32/8000	<name>_g726-32.<ext>
G.729	g729/8000	<name>_g729.<ext>
G729a	g729/8000	<name>_g729a.<ext>
G729d	g729d/8000	<name>_g729d.<ext>
G729e	g729e/8000	<name>_g729e.<ext>
GSM Full Rate (GSM 6.1.0)	gsm/8000	<name>_gsm. or <name>_gsmFR.<ext>
MS-GSM (Microsoft GSM)	—	<name>_msgsm.<ext> or <name>_gsmF.<ext>

# Deploying SIP Feature Server

Complete these tasks to configure SIP Feature Server instances for standalone operation.

1. [Configure SIP Feature Server applications](#)
2. [Deploy co-located/external Cassandra cluster](#)
3. [Configure SIP Feature Server for co-located/external Cassandra cluster](#)
4. [Implement the Feature Server GAX Plug-in](#)
5. [Implement device management](#)
6. [Start and verify SIP Feature Server](#)
7. [Configure SIP Server for Feature Server](#)
8. [Configure voicemail](#)

## Cassandra Cluster Tasks

- [Add a New Data Center](#)
- [Configure Regional Voicemail Storage](#)

You can use [these sample deployments](#) as a model for your chosen deployment option.

## Videos

The following videos provide an overview and procedure of the deployment process. For detailed information, refer to the other topics covered in this section.

### Deployment Process for Windows

#### Installing Feature Server Part I/III

[Link to video](#)

#### Installing Feature Server Part II/III

[Link to video](#)

#### Installing Feature Server Part III/III

[Link to video](#)

---

## Deployment Process for Linux

Installing Feature Server Part I/III

[Link to video](#)

Installing Feature Server Part II/III

[Link to video](#)

Installing Feature Server Part III/III

[Link to video](#)

---

# Configure SIP Feature Server applications

Your Feature Server deployment differs depending on the deployment option you choose: single-site, multisite (with one or multiple data centers), or Business Continuity (single-site or multisite). If you haven't yet decided, [review the options](#).

Feature Server automatically installs Apache Cassandra as its primary database management system. While most Cassandra configuration occurs during installation, you must set some values to match your environment.

These instructions include the Cassandra options specific to Feature Server. To learn more about Cassandra, see the [Cassandra wiki](#) and the DataStax [Cassandra documentation website](#).

## Important

After SIP Feature Server is started for the first time, the option **[Cluster].disable-initial-import** must be configured and set to true. For more details, refer to [Configuration Options](#).

## Single-site

To deploy Feature Server on a single site, you deploy two Feature Server instances configured as an active-active HA (High Availability) pair, running with a single SIP switch and SIP Server. See [Deployment Options](#) for architectural details.

### To deploy SIP Feature Server on a single site:

1. Ensure that you have met all the [installation prerequisites](#) for SIP Feature Server. If you assign port numbers, ensure that you do not use any of the [reserved ports](#).
2. Configure the SIP Feature Server applications: In Genesys Administrator, use the supplied template (**Templates > GenesysSIPFeatureServer\_812.apd**) to create an application. Note the Application object name, which the application installer requires.
3. Create a Host for the target machine.
4. If this is the first Feature Server instance you are deploying:
  - Designate it to handle the initial synchronization of data from Configuration Server to all Feature Servers in a tenant. In the **Cluster > Options** tab of the Feature Server Application in Genesys Administrator, set master to true. See [Configuration options](#).
  - In the **Cassandra** section:
    - set **replicationOptions** to DC1=2
    - set **replicationStrategyClassName** to NetworkTopologyStrategy
5. For new deployments, set **cassandra-counter** to false in the **VoicemailServer** section. SIP Feature

---

Server uses Cassandra counters to maintain the MWI count for the mailboxes. The Cassandra counters are not stable and may result in inconsistency in the MWI count. When **cassandra-counter** is set to false, Feature Server maintains consistency in the MWI count at all times.

6. For the other Feature Server, in the **Cluster** section of the Feature Server Application, set master to false and confsync to false.  
**Important:** Ensure that you designate only one Feature Server instance as the master.
  7. In the Feature Server Connections tab, add the SIP Server to the Feature Server. Ensure that the PortID is the default SIP Server port.
  8. Ensure that your Configuration Server history log settings are large enough to accommodate all configuration changes that occur whenever you simultaneously shut down your master Feature Server and your confsync-enabled Feature Server (note that such shutdowns are not recommended; try to leave at least one confsync-enabled server running at all times). In the **[history-log]** section of the Configuration Server application object:
    - set **expiration** to at least the maximum number of days you expect your servers to be offline
    - set **max-records** to the maximum number of configuration records you expect your servers to handle during the offline period

For option details, see [Configuration Server Configuration Options](#).
  9. Review the current [known issues and recommendations](#) for this release.
  10. To install Feature Server, run `install.sh` (Linux) or `setup.exe` (Windows) from the product DVD. Follow the installer prompts, using the default values except for:
    - Set the deployment mode to Standalone.
    - Supply the Cassandra cluster name (use the same name for all Cassandra instances in the Cassandra cluster).
    - For Seeds, supply the IP address of the master Feature Server.
    - Supply a Cassandra storage location, but do **not** use the installation directory.
  11. Set up the co-located/external Cassandra cluster by referring the following articles:
    - [Deploy co-located/external Cassandra cluster](#)
    - [Configure SIP Feature Server for co-located/external Cassandra cluster](#).
  12. If this is the master Feature Server, disable the server firewall.
  13. Repeat the above steps to create the non-master Feature Server, keeping in mind the differences between the master instance and the other instance.
  14. If your environment includes **more than two** Feature Server instances per SIP switch, create an FQDN that includes the IP addresses of all the **N** Feature Server instances (the total number of instances minus 1), then set up a DNS server (your choice) to distribute the workload if the first instance goes down.

For example, if 172.24.128.34, 172.24.128.35, and 172.24.128.36 are the IP addresses of three instances, you create the FQDN with any of the two instances: FQDN1 = `fs.genesys.com`, which resolves to 172.24.128.34 and 172.24.128.35. You use the IP address of the "extra" instance as the *FS(n+1) IP address*, and the *FQDN1* value, during the [Configure voicemail](#) and [Implement device management](#) procedures.
  15. To enable a Transport Layer Security autodetect (upgradable) connection, configure an autodetection port. See Introduction to Genesys Transport Layer Security in [Genesys 8.1 Security Deployment Guide](#).
  16. Set the remaining [Feature Server configuration options](#).
-

## Appendix

Use the following configuration if you're using an Embedded Cassandra cluster in your SIP Feature Server deployment. However, note that this cluster mode is being deprecated beginning with version 8.1.203.xx and will no longer be supported in future versions.

To configure an embedded Cassandra cluster, follow all the steps from the above procedure but replace the Step-11 configuration with the following:

1. Edit the file ***Cassandra storage location/etc/Cassandra.yaml***:
  - Verify that the file contains these properties and values, and update as needed (do not change any other values):
    - `cluster_name`: 'FeatureServerCluster' (or whatever name you supplied as the Cassandra cluster name during installation)
    - `data_file_directories`: *Cassandra storage location/storage/data*
    - `commitlog_directory`: *Cassandra storage location/storage/commitlog*
    - `endpoint_snitch`: PropertyFileSnitch/GossipingPropertyFileSnitch
    - `listen_address`: *IP address or FQDN of the instance that you are configuring*
    - `saved_caches_directory`: *Cassandra storage location/storage/saved\_caches*
    - `storage_port`: 7000 (set each Cassandra node to 7000)
    - `-seeds`: *master Feature Server IP address or FQDN*  
Cassandra nodes use the seed node list to find each other and learn the topology of the ring.
2. For PropertyFileSnitch, in the Feature Server home/resources folder, create or edit the **`cassandra-topology.properties`** file and add the following, where the data center names are the names you specified in the **`replicationOptions`** configuration option earlier:
  - `#Cassandra Node IP=Data Center:Rack`
  - IP or FQDN of the FS1 host=data center 1 name:RAC1
  - IP or FQDN of the FS2 host=data center 1 name:RAC2
3. For GossipingPropertyFileSnitch, in the Feature Server home/resources folder, create or edit the **`cassandra-rackdc.properties`** file in each node and add data center and rack information specific to that node, where the data center names are the names you specified in the **`replicationOptions`** configuration option earlier:
  - `dc`=name of data center to which the specific node belongs
  - `rack`=name of rack to which the specific node belongs

## Multisite (single DC)

You can deploy Feature Server in a multisite (multi-switch) environment, using one or multiple data centers:

- Use a single data center if your sites are co-located.
-

- Use multiple data centers, one for each site, if your sites are in different physical locations.

Multisite configurations require a minimum of four Feature Server instances, two for each site.

### To deploy SIP Feature Server in a multisite, single-data center environment:

1. Ensure that you have met all the [installation prerequisites](#) for SIP Feature Server. If you assign port numbers, ensure that you do not use any of the [reserved ports](#).
2. Configure the SIP Feature Server applications: In Genesys Administrator, use the supplied template (**Templates > GenesysSIPFeatureServer\_812.apd**) to create an application. Note the Application object name, which the application installer requires.
3. Create a Host for the target machine.
4. If this is the first Feature Server instance you are deploying:
  - Designate it to handle the initial synchronization of data from Configuration Server to all Feature Servers in a tenant. In the **Cluster > Options** tab of the Feature Server Application in Genesys Administrator, set master to true. See [Configuration options](#).
  - In the **Cassandra** section:
    - set **replicationOptions** to DC1=3 or more, depending on the number of nodes in the cluster: increment the value by one for each node you add beyond 4.
    - set **replicationStrategyClassName** to NetworkTopologyStrategy
5. For new deployments, set **cassandra-counter** to false in the **VoicemailServer** section. SIP Feature Server uses Cassandra counters to maintain the MWI count for the mailboxes. The Cassandra counters are not stable and may result in inconsistency in the MWI count. When **cassandra-counter** is set to false, Feature Server maintains consistency in the MWI count at all times.
6. On **one** other Feature Server instance on each site, in the **Cluster** section of the Feature Server Application, set master to false and confsync to true. On **all** other instances, set master to false and confsync to false.  
**Important:** Ensure that you designate only one Feature Server instance as the master.
7. In the Feature Server Connections tab, add the SIP Server to the Feature Server. Ensure that the PortID is the default SIP Server port.
8. Ensure that your Configuration Server history log settings are large enough to accommodate all configuration changes that occur whenever you simultaneously shut down your master Feature Server and your confsync-enabled Feature Server (note that such shutdowns are not recommended; try to leave at least one confsync-enabled server running at all times). In the **[history-log]** section of the Configuration Server application object:
  - set **expiration** to at least the maximum number of days you expect your servers to be offline
  - set **max-records** to the maximum number of configuration records you expect your servers to handle during the offline period

For option details, see [Configuration Server Configuration Options](#).
9. Review the current [known issues and recommendations](#) for this release.
10. To install Feature Server, run `install.sh` (Linux) or `setup.exe` (Windows) from the product DVD. Follow the installer prompts, using the default values except for:
  - Set the deployment mode to Standalone.
  - Supply the Cassandra cluster name (use the same name for all Cassandra instances in the

---

Cassandra cluster).

- For Seeds, supply the IP address of the master Feature Server.
  - Supply a Cassandra storage location, but do **not** use the installation directory.
11. Set up the co-located/external Cassandra cluster by referring the following articles:
    - [Deploy co-located/external Cassandra cluster](#)
    - [Configure SIP Feature Server for co-located/external Cassandra cluster](#).
  12. If this is the master Feature Server, disable the server firewall.
  13. Repeat the above steps for each Feature Server instance you want to create (at least four total), keeping in mind the differences between the master instance and the other instances.
  14. If your environment includes **more than two** Feature Server instances per SIP switch, create an FQDN that includes the IP addresses of all the **N** Feature Server instances (the total number of instances minus 1), then set up a DNS server (your choice) to distribute the workload if the first instance goes down.

For example, if 172.24.128.34, 172.24.128.35, and 172.24.128.36 are the IP addresses of three instances, you create the FQDN with any of the two instances: FQDN1 = fs.genesys.com, which resolves to 172.24.128.34 and 172.24.128.35. You use the IP address of the "extra" instance as the *FS(n+1)* IP address, and the *FQDN1* value, during the [Configure voicemail](#) and [Implement device management](#) procedures.
  15. To enable a Transport Layer Security autodetect (upgradable) connection, configure an autodetection port. See Introduction to Genesys Transport Layer Security in [Genesys 8.1 Security Deployment Guide](#).
  16. Set the remaining [Feature Server configuration options](#).

## Appendix

Use the following configuration if you're using an Embedded Cassandra cluster in your SIP Feature Server deployment. However, note that this cluster mode is being deprecated beginning with version 8.1.203.xx and will no longer be supported in future versions.

To configure an embedded Cassandra cluster, follow all the steps from the above procedure but replace the Step-11 configuration with the following:

1. Edit the file ***Cassandra storage location/etc/Cassandra.yaml***:
  - Verify that the file contains these properties and values, and update as needed (do not change any other values):
    - `cluster_name`: 'FeatureServerCluster' (or whatever name you supplied as the Cassandra cluster name during installation)
    - `data_file_directories`: *Cassandra storage location/storage/data*
    - `commitlog_directory`: *Cassandra storage location/storage/commitlog*
    - `endpoint_snitch`: PropertyFileSnitch/GossipingPropertyFileSnitch
    - `listen_address`: *IP address or FQDN of the instance that you are configuring*
    - `saved_caches_directory`: *Cassandra storage location/storage/saved\_caches*
    - `storage_port`: 7000 (set each Cassandra node to 7000)
    - `-seeds`: *master Feature Server IP address or FQDN*

Cassandra nodes use the seed node list to find each other and learn the topology of the ring.

2. For `PropertyFileSnitch`, in Feature Server home/resources folder, create or edit the **cassandra-topology.properties** file and add the following, where the data center names are the names you specified in the **replicationOptions** configuration option earlier:
  - `#Cassandra Node IP=Data Center:Rack`
  - `IP or FQDN of the FS1 host=data center 1 name:RAC1`
  - `IP or FQDN of the FS2 host=data center 1 name:RAC2`
3. For `GossipingPropertyFileSnitch`, in the Feature Server home/resources folder, create or edit the **cassandra-rackdc.properties** file in each node and add data center and rack information specific to that node, where the data center names are the names you specified in the **replicationOptions** configuration option earlier:
  - `dc=name of data center to which the specific node belongs`
  - `rack=name of rack to which the specific node belongs`

## Multisite (multiple DCs)

You can deploy Feature Server in a multisite (multi-switch) environment, using one or multiple data centers:

- Use a single data center if your sites are co-located.
- Use multiple data centers, one for each site, if your sites are in different physical locations.

Multisite configurations require a minimum of four Feature Server instances, two for each site. In multi-data center mode, each site contains its own data center.

### To deploy SIP Feature Server in a multisite, multiple-data center environment:

1. Ensure that you have met all the [installation prerequisites](#) for SIP Feature Server. If you assign port numbers, ensure that you do not use any of the [reserved ports](#).
2. Configure the SIP Feature Server applications: In Genesys Administrator, use the supplied template (**Templates > GenesysSIPFeatureServer\_812.apd**) to create an application. Note the Application object name, which the application installer requires.
3. Create a Host for the target machine.
4. If this is the first Feature Server instance you are deploying:
  - Designate it to handle the initial synchronization of data from Configuration Server to all Feature Servers in a tenant. In the **Cluster > Options** tab of the Feature Server Application in Genesys Administrator, set master to true. See [Configuration options](#).
  - In the **Cassandra > Options** tab:
    - set **replicationOptions** to `data center 1 name=2,data center 2 name=2` (remember these data center names to use in the `cassandra-topology.properties` file you create or edit below)
    - set **replicationStrategyClassName** to `NetworkTopologyStrategy`

- 
5. For new deployments, set **cassandra-counter** to false in the **VoicemailServer** section. SIP Feature Server uses Cassandra counters to maintain the MWI count for the mailboxes. The Cassandra counters are not stable and may result in inconsistency in the MWI count. When **cassandra-counter** is set to false, Feature Server maintains consistency in the MWI count at all times.
  6. On **one** other Feature Server instance on each site, in the **Cluster** section of the Feature Server Application, set master to false and confsync to true. On **all** other instances, set master to false and confsync to false.  
**Important:** Ensure that you designate only one Feature Server instance as the master.
  7. In the Feature Server Connections tab, add the SIP Server to the Feature Server. Ensure that the PortID is the default SIP Server port.
  8. Ensure that your Configuration Server history log settings are large enough to accommodate all configuration changes that occur whenever you simultaneously shut down your master Feature Server and your confsync-enabled Feature Server (note that such shutdowns are not recommended; try to leave at least one confsync-enabled server running at all times). In the **[history-log]** section of the Configuration Server application object:
    - set **expiration** to at least the maximum number of days you expect your servers to be offline
    - set **max-records** to the maximum number of configuration records you expect your servers to handle during the offline periodFor option details, see [Configuration Server Configuration Options](#).
  9. Review the current [known issues and recommendations](#) for this release.
  10. To install Feature Server, run `install.sh` (Linux) or `setup.exe` (Windows) from the product DVD. Follow the installer prompts, using the default values except for:
    - Set the deployment mode to Standalone.
    - Supply the Cassandra cluster name (use the same name for all Cassandra instances in the Cassandra cluster).
    - For Seeds, supply the IP address of the master Feature Server.
    - Supply a Cassandra storage location, but do **not** use the installation directory.
  11. Set up the co-located/external Cassandra cluster by referring the following articles:
    - [Deploy co-located/external Cassandra cluster](#)
    - [Configure SIP Feature Server for co-located/external Cassandra cluster](#).
  12. If this is the master Feature Server, disable the server firewall.
  13. Repeat the above steps for each Feature Server instance you want to create, keeping in mind the differences between the master instance and the other instances.
  14. In an environment with more than two Feature Server instances, you must create an FQDN that includes the IP addresses of all the **N** Feature Server instances (the total number of instances minus 1), then set up a DNS server (your choice) to distribute the workload if the first instance goes down.  
For example, if 172.24.128.34, 172.24.128.35, and 172.24.128.36 are the IP addresses of three instances, you create the FQDN with any of the two instances: `FQDN1 = fs.genesys.com`, which resolves to 172.24.128.34 and 172.24.128.35. You use the IP address of the "extra" instance as the *FS(n+1) IP address*, and the *FQDN1* value, during the [Configure voicemail](#) and [Implement device management](#) procedures.
  15. To enable a Transport Layer Security autodetect (upgradable) connection, configure an autodetection port. See Introduction to Genesys Transport Layer Security in [Genesys 8.1 Security Deployment Guide](#).
-

16. Set the remaining **Feature Server configuration options**.

## Appendix

Use the following configuration if you're using an Embedded Cassandra cluster in your SIP Feature Server deployment. However, note that this cluster mode is being deprecated beginning with version 8.1.203.xx and will no longer be supported in future versions.

To configure an embedded Cassandra cluster, follow all the steps from the above procedure but replace the Step-11 configuration with the following:

1. Edit the file ***Cassandra storage location/etc/Cassandra.yaml***:

- Verify that the file contains these properties and values, and update as needed (do not change any other values):
  - `cluster_name`: 'FeatureServerCluster' (or whatever name you supplied as the Cassandra cluster name during installation)
  - `data_file_directories`: *Cassandra storage location/storage/data*
  - `commitlog_directory`: *Cassandra storage location/storage/commitlog*
  - `endpoint_snitch`: PropertyFileSnitch/GossipingPropertyFileSnitch
  - `listen_address`: *IP address or FQDN of the instance that you are configuring*
  - `saved_caches_directory`: *Cassandra storage location/storage/saved\_caches*
  - `storage_port`: 7000 (set each Cassandra node to 7000)
  - `-seeds`: *master Feature Server IP address or FQDN*  
Cassandra nodes use the seed node list to find each other and learn the topology of the ring.

2. For PropertyFileSnitch, in the folder ***Feature Server home/resources***, create or edit a file called `cassandra-topology.properties` and add the following, where the data center names are the names you specified in the **replicationOptions** configuration option above:

- `# Cassandra Node IP=Data Center:Rack`
- `IP or FQDN of the FS1 host=data center 1 name:RAC1`
- `IP or FQDN of the FS2 host=data center 1 name:RAC2`
- `IP or FQDN of the FS3 host=data center 2 name:RAC1`
- `IP or FQDN of the FS4 host=data center 2 name:RAC2`
- `# default for unknown nodes`
- `default=data center 1 name:RAC1`

3. For GossipingPropertyFileSnitch, in the Feature Server home/resources folder, create or edit the **`cassandra-rackdc.properties`** file in each node and add data center and rack information specific to that node, where the data center names are the names you specified in the **replicationOptions** configuration option earlier:

- `dc`=name of data center to which the specific node belongs
- `rack`=name of rack to which the specific node belongs

---

## Business Continuity

Though you can achieve some degree of Business Continuity by deploying Feature Server in single-data center mode, the recommended deployment uses multiple data centers, which provides disaster recovery. All Business Continuity deployments must contain at least two Feature Server instances per site.

### To deploy SIP Feature Server in Business Continuity mode:

1. Ensure that you have met all the [installation prerequisites](#) for SIP Feature Server. If you assign port numbers, ensure that you do not use any of the [reserved ports](#).
2. Configure the SIP Feature Server applications: In Genesys Administrator, use the supplied template (**Templates > GenesysSIPFeatureServer\_812.apd**) to create an application. Note the Application object name, which the application installer requires.
3. Create a Host for the target machine.
4. If this is the first Feature Server instance you are deploying:
  - Designate it to handle the initial synchronization of data from Configuration Server to all Feature Servers in a tenant. In the **Cluster > Options** tab of the Feature Server Application in Genesys Administrator, set master to true. See [Configuration options](#).
  - In the **Cassandra > Options** tab:
    - set **replicationOptions** to *data center 1 name=2,data center 2 name=2* (remember these data center names to use in the `cassandra-topology.properties` file you create or edit below)
    - set **replicationStrategyClassName** to `NetworkTopologyStrategy`
5. For new deployments, set **cassandra-counter** to false in the **VoicemailServer** section. SIP Feature Server uses Cassandra counters to maintain the MWI count for the mailboxes. The Cassandra counters are not stable and may result in inconsistency in the MWI count. When **cassandra-counter** is set to false, Feature Server maintains consistency in the MWI count at all times.
6. On **one** other Feature Server instance on each site, in the **Cluster** section of the Feature Server Application, set master to false and confsync to true. On **all** other instances, set master to false and confsync to false.  
**Important:** Ensure that you designate only one Feature Server instance as the master.
7. In the Feature Server Connections tab, add the SIP Server to the Feature Server. Ensure that the PortID is the default SIP Server port.
8. Ensure that your Configuration Server history log settings are large enough to accommodate all configuration changes that occur whenever you simultaneously shut down your master Feature Server and your confsync-enabled Feature Server (note that such shutdowns are not recommended; try to leave at least one confsync-enabled server running at all times). In the **[history-log]** section of the Configuration Server application object:
  - set **expiration** to at least the maximum number of days you expect your servers to be offline
  - set **max-records** to the maximum number of configuration records you expect your servers to handle during the offline period

For option details, see [Configuration Server Configuration Options](#).
9. Review the current [known issues and recommendations](#) for this release.
10. To install Feature Server, run `install.sh` (Linux) or `setup.exe` (Windows) from the product DVD. Follow

---

the installer prompts, using the default values except for:

- Set the deployment mode to Standalone.
  - Supply the Cassandra cluster name (use the same name for all Cassandra instances in the Cassandra cluster).
  - For Seeds, supply the IP address of the master Feature Server.
  - Supply a Cassandra storage location, but do **not** use the installation directory.
11. Set up the co-located/external Cassandra cluster by referring the following articles:
    - [Deploy co-located/external Cassandra cluster](#)
    - [Configure SIP Feature Server for co-located/external Cassandra cluster](#)
  12. If this is the master Feature Server, disable the server firewall.
  13. Repeat the above steps for each Feature Server instance you want to create, keeping in mind the differences between the master instance and the other instances.
  14. In an environment with more than two Feature Server instances, you must create an FQDN that includes the IP addresses of all the **N** Feature Server instances (the total number of instances minus 1), then set up a DNS server (your choice) to distribute the workload if the first instance goes down.  
For example, if 172.24.128.34, 172.24.128.35, and 172.24.128.36 are the IP addresses of three instances, you create the FQDN with any of the two instances: FQDN1 = fs.genesys.com, which resolves to 172.24.128.34 and 172.24.128.35. You use the IP address of the "extra" instance as the *FS(n+1)* IP address, and the *FQDN1* value, during the [Configure voicemail](#) and [Implement device management](#) procedures.
  15. To enable a Transport Layer Security autodetect (upgradable) connection, configure an autodetection port. See Introduction to Genesys Transport Layer Security in [Genesys 8.1 Security Deployment Guide](#).
  16. Set the remaining [Feature Server configuration options](#).

## Appendix

Use the following configuration if you're using an Embedded Cassandra cluster in your SIP Feature Server deployment. However, note that this cluster mode is being deprecated beginning with version 8.1.203.xx and will no longer be supported in future versions.

To configure an embedded Cassandra cluster, follow all the steps from the above procedure but replace the Step-11 configuration with the following:

1. Edit the file ***Cassandra storage location/etc/Cassandra.yaml***:
  - Verify that the file contains these properties and values, and update as needed (do not change any other values):
    - `cluster_name`: 'FeatureServerCluster' (or whatever name you supplied as the Cassandra cluster name during installation)
    - `data_file_directories`: *Cassandra storage location/storage/data*
    - `commitlog_directory`: *Cassandra storage location/storage/commitlog*
    - `endpoint_snitch`: PropertyFileSnitch/GossipingPropertyFileSnitch
    - `listen_address`: *IP address or FQDN of the instance that you are configuring*
    - `saved_caches_directory`: *Cassandra storage location/storage/saved\_caches*

- `storage_port: 7000` (set each Cassandra node to 7000)
  - `-seeds: master Feature Server IP address or FQDN`  
Cassandra nodes use the seed node list to find each other and learn the topology of the ring.
2. For `PropertyFileSnitch`, in the folder **Feature Server home/resources**, create or edit a file called `cassandra-topology.properties` and add the following, where the data center names are the names you specified in the **replicationOptions** configuration option above:
- `# Cassandra Node IP=Data Center:Rack`
  - `IP or FQDN of the FS1 host=data center 1 name:RAC1`
  - `IP or FQDN of the FS2 host=data center 1 name:RAC2`
  - `IP or FQDN of the FS3 host=data center 2 name:RAC1`
  - `IP or FQDN of the FS4 host=data center 2 name:RAC2`
  - `# default for unknown nodes`
  - `default=data center 1 name:RAC1`
  - For `GossipingPropertyFileSnitch`, in the Feature Server home/resources folder, create or edit the **`cassandra-rackdc.properties`** file in each node and add data center and rack information specific to that node, where the data center names are the names you specified in the **replicationOptions** configuration option earlier:
    - `dc=name of data center to which the specific node belongs`
    - `rack=name of rack to which the specific node belongs`
-

---

# Deploy a co-located/external Cassandra cluster

The Apache Cassandra database is an open source NoSQL database, which is easily scalable and provides high availability without compromising performance.

## Important

SIP Feature Server version to be installed is 8.1.201.91 or higher.

SIP Feature Server with co-located/external Cassandra cluster is supported only for new deployments. A co-located/external Cassandra cluster should be deployed with Cassandra version 2.2.8 or higher.

Java Runtime Environment (JRE) v1.8 64-bit or higher is required for Cassandra 3.x.

Ensure that you have a minimum of two nodes per data center and the clocks on all Cassandra nodes are synchronized.

## Prerequisites to use external Cassandra cluster

1. The external Cassandra cluster deployed can be co-located/shared with other Genesys components. However, the SIP Feature Server keyspaces (global and regional) should not be shared with other components.
2. The minimum disk space required for SIP Feature Server keyspaces can be computed using the disk sizing tool mentioned in the [Hardware and software prerequisites](#) page.
3. While sharing Cassandra with multiple components, disk requirement is computed by summing the minimum required space from all components.

## Selecting a Seed node

Seed node is a comma-delimited list of IP addresses used by gossip for bootstrapping new nodes joining a cluster. In multiple data-center clusters, the seed list must include at least one node from each data center (replication group). More than a single seed node per data center is recommended for fault tolerance. Otherwise, gossip has to communicate with another data center when bootstrapping a node. Making every node a seed node is not recommended because of increased maintenance and reduced gossip performance. Gossip optimization is not critical, but Genesys recommends that you use a small seed list. Usually, the first node installed in each data center is considered as a seed node. For more information on seed nodes and gossip, see [Internode communications \(gossip\)](#).

## Deploying Cassandra cluster

The following steps show how to deploy each Cassandra node in the co-located/external Cassandra

---

cluster:

1. Download the latest version in Apache Cassandra 2.2 or in 3.x from either the [Cassandra archive index](#) page or from the [Downloading Cassandra](#) page.
2. Extract the downloaded zip file to any desired directory (Cassandra installed directory).
3. Edit the **<Cassandra installed directory>\conf\cassandra.yaml** file and configure the parameters:
  - `cluster_name` : FeatureServerCluster
  - `start_rpc` : true
  - `listen_address` : IP or FQDN of the node
  - `rpc_address` : IP or FQDN of the node
  - `seeds` : comma separated IP or FQDN of the seed nodes
  - `storage_port` : 7000 (default value)
  - `ssl_storage_port` : 7001 (default value)
  - `native_transport_port`: 9042 (default value)
  - `rpc_port` : 9160 (default value)
  - `endpoint_snitch` : PropertyFileSnitch
4. Edit the **<Cassandra installed directory>\conf\cassandra-topology.properties** file and configure the data centers as follows:
  - The following example shows a single data center with two Cassandra nodes:
    - Cassandra node 1 IP or FQDN 1=data center 1 name:RAC1
    - Cassandra node 2 IP or FQDN 2=data center 1 name:RAC2
  - The following example shows a multi data center with four Cassandra nodes with two nodes per data centre:
    - Cassandra node 1 IP or FQDN 1=data center 1 name:RAC1
    - Cassandra node 2 IP or FQDN 2=data center 1 name:RAC2
    - Cassandra node 3 IP or FQDN 3=data center 2 name:RAC1
    - Cassandra node 4 IP or FQDN 4=data center 2 name:RAC2

## Configuring Cassandra logging

By default, the Cassandra logs are generated under **<Cassandra installed directory>\logs**. To change the log file directory:

- On Linux, edit the **<Cassandra installed directory>\bin\cassandra** file and update the parameter:
  - `-Dcassandra.logdir=$CASSANDRA_HOME/logs`
- On Windows, edit the **<Cassandra installed directory>\bin\cassandra.bat** file and update the parameter:
  - `Dcassandra.logdir="%CASSANDRA_HOME%\logs`

---

To configure Cassandra logging, see [Configuring logging](#).

## Configuring Cassandra Authentication and Authorization

1. On each Cassandra node, edit the **<Cassandra installed directory>\conf\cassandra.yaml** file and set the following parameters and restart the nodes:
    - `authenticator :PasswordAuthenticator`
    - `authorizer :CassandraAuthorizer`
  2. On the Master Cassandra node, navigate to **<Cassandra installed directory>\bin\** and run the CQL client as follows:
    - On Linux
      - `./cqlsh <IP or FQDN of the Cassandra node configured in Cassandra.yaml> <CQL Port>`
    - On Windows
      - `cqlsh <IP or FQDN of the Cassandra node configured in Cassandra.yaml> <CQL Port>`
    - By default, the CQL Port is 9042.
  3. Increase the replication factor of the “system\_auth”(Pre-defined keyspace in Cassandra 2.2 or higher) keyspace by using the CQL query:
    - `alter keyspace system_auth with replication = {'class': 'NetworkTopologyStrategy', <replication factor>};`  
For example, if Cassandra cluster is configured with two Data centers (DC1 & DC2) and each data center is configured with two nodes, then set the replication factor of system\_auth keyspace as DC1:2, DC2:2 by using the cql query as follows:
      - `alter keyspace system_auth with replication = {'class': 'NetworkTopologyStrategy', 'DC1': 2, 'DC2': 2};`
  4. Create a user using the following cql query:
    - `create user <user_name> with password <password>;`
  5. Create a role by using the following cql query:
    - `create role <role_name>;`
  6. SIP Feature Server must be authorized to create and access keyspace, hence grant all permissions to the role name
    - `grant all permissions on all keyspaces to <role_name>;`
  7. Grant access to the created user.
    - `grant <role_name> to <user_name>;`
  8. If the Cassandra cluster is used for any other Genesys components, revoke the access rights after master SIP Feature Server is started for the first time, and grant all permissions only to the SIP Feature Server keyspace (sipfs) by using the following query:
    - `revoke all permissions on all keyspaces from <role_name>;`
    - `grant all permissions on keyspace sipfs to <role_name>;`
-

9. If regional keyspace is created, then grant permissions to regional keyspace as well.
  - grant all permissions on keyspace <regional keyspace name> to <role\_name>;

## Configuring Cassandra SSL

The following steps show how to enable secure connection (SSL) for each Cassandra node:

1. Generate server certificates and keystore. For more information on certificate generation and installation, see [Genesys Security Deployment Guide](#).
2. Copy the keystore file generated during certificate installation to the **<Cassandra Installed directory>\bin**.
3. Update the **Client-to-node** option in the **Cassandra.yaml** file as follows:

```
client_encryption_options
  enabled: true

  optional: false

  keystore: <keystore_name>

  keystore_password: <keystore_password>
```

4. Update the **node-to-node** option in the **Cassandra.yaml** file as follows:

```
internode_encryption: all
  keystore: <keystore name>

  keystore_password: <keystore_password>

  truststore: <truststore name>

  truststore_password: <truststore_password>

  require_client_auth: true|false

  protocol: (Default: TLS)
```

5. Restart Cassandra node.

## Monitoring Cassandra

### Linux

#### Configuring Cassandra as a Service

1. Create the `/etc/init.d/cassandra` startup script.
2. Edit the contents of the file:

```
#!/bin/sh
#
# chkconfig: - 80 45
# description: Starts and stops Cassandra
```

```
# update daemon path to point to the cassandra executable
DAEMON=<Cassandra installed directory>/bin/cassandra
start() {
    echo -n "Starting Cassandra... "
    $DAEMON -p /var/run/cassandra.pid
    echo "OK"
    return 0
}
stop() {
    echo -n "Stopping Cassandra... "
    kill $(cat /var/run/cassandra.pid)
    echo "OK"
    return 0
}
case "$1" in
    start)
        start
        ;;
    stop)
        stop
        ;;
    restart)
        stop
        start
        ;;
    *)
        echo $"Usage: $0 {start|stop|restart}"
        exit 1
esac
exit $?
```

### 3. Make the file executable:

```
sudo chmod +x /etc/init.d/cassandra
```

### 4. Add the new service to the list:

```
sudo chkconfig --add cassandra
```

### 5. Now you can manage the service from the command line:

```
sudo /etc/init.d/cassandra start
sudo /etc/init.d/cassandra stop
```

## Start/Stop Cassandra Cluster

Start the seed node and then start the rest of the Cassandra nodes sequentially:

```
Start: sudo service cassandra start
Stop: sudo service cassandra stop
```

## Verifying Cassandra Cluster

After you have deployed Cassandra Cluster, you may want to verify that all the nodes can communicate with each other. To do this, execute the following command on any Cassandra host:

```
cd <Cassandra installed directory>/bin
./nodetool status
```

## Windows

### Monitoring Cassandra as a service

Currently this is supported if Cassandra is deployed in a windows machine. First we need configure Cassandra to run as a service and then configure the service as third party application in GA to control using SCI.

### Configuring Cassandra as a Service

1. Download the latest apache commons daemon from [Apache Commons Project Distributions](#).
2. Extract the commons daemon in **<Cassandra installed directory>\bin**.
3. Rename the extracted folder as daemon.
4. Add <Cassandra installed directory> as CASSANDRA\_HOME in windows environment variable.
5. Edit the **cassandra.yaml** file in **<Cassandra installed directory>\conf** and uncomment the `data_file_directories`, `commitlog_directory`, `saved_cache_directory` and set the absolute paths.
6. Edit **cassandra.bat** in **<Cassandra installed directory>\bin** and replace the value for the `PATH_PRUNSRV` as follows:
  - for 32 bit windows, set `PATH_PRUNSRV=%CASSANDRA_HOME%\bin\daemon\`
  - for 64 bit windows, set `PATH_PRUNSRV=%CASSANDRA_HOME%\bin\daemon\amd64\`
7. Change the service name for `SERVICE_JVM` as required in **cassandra.bat**.
  - For example, setting `SERVICE_JVM="cassandra_228"` creates the windows service in the `cassandra_228` name.
8. With administrator privileges, run **cassandra.bat install** from command prompt.

This creates a Windows Service.

### Configure Cassandra Service in GA

1. Create a new application object with application template **ThirdPartyServer**.
  2. Under Server Info, provide the host in which the Cassandra node is running.
  3. Under Start Info, provide the working directory as:
    - for 32 bit windows, set `<Cassandra installed directory>\bin\daemon\`
    - for 64 bit windows, set `<Cassandra installed directory>\bin\daemon\amd64\`
  4. Under Start info, provide the command line as:
    - `prunsvr.exe`
-

5. Under Start info, provide the command line arguments as:
  - //RS//<service name>
6. In the **Annex** tab, create a new section called **start\_stop**.
7. In the **start\_stop** section, create an option **start\_command** with value `net start <cassandra service>`.
8. In the **start\_stop** section, create an option **stop\_command** with value `net stop <cassandra service>`.

Now, monitor (start/stop) Cassandra from SCI.

## Start/Stop Cassandra Cluster

1. Start the seed node first, followed by rest of the Cassandra nodes sequentially.
  - Use SCI to start/stop the Cassandra nodes as mentioned in the Monitoring Cassandra as a service on Windows section earlier.

## Verifying Cassandra Cluster

After you have deployed the Cassandra Cluster, you can verify that all the nodes communicate with each other. To do this, run the following command on any Cassandra host:

```
cd <Cassandra installed directory>\bin
nodetool status
```

## Deactivating Embedded Cassandra module for version 8.1.203 and later

If you have upgraded SIP Feature Server to version 8.1.203 and later and used external Cassandra in your deployment, you can deactivate the Embedded Cassandra module from the deployment. Note that deactivating the Embedded Cassandra module is recommended but it is an optional step.

To deactivate the Embedded Cassandra module:

1. Locate the **start.ini** file in the path: **<FS installation folder>/start.ini**.
2. Open the file with a text editor and remove the line **--module=fs-cass11**.
3. Save the file.
4. Restart SIP Feature Server if it is running.
5. After restart, remove the installation files from the **<FS installation folder>/lib/fs-cass11** folder.

## Configure SIP Feature Server for a co-located/external Cassandra cluster

Perform the following steps on each SIP Feature Server instance:

1. If you are using Feature Server version 8.1.201.92 or above, then during installation, select Cassandra cluster type as **External Cassandra** cluster type. If you are using SIP Feature Server version 8.1.201.91, then edit the **<Feature Server installed directory>\launcher.xml** file and set the following parameters:
  - `com.genesyslab.common.application.cassandraServer=false`
2. In the **SIP Feature Server application > [Cassandra]** section, configure the options as follows:
  - `nodes=IPAddress` of all the Cassandra nodes that are available in that data center.
  - `keyspace=<keyspace_name>` Keyspace name for SIP Feature Server application. The default is `sipfs`.
  - `nodeFailureTolerance=replication_factor` Value of its data center-1.  
For example, if DC1 is the data center where SIP Feature Server is connected and the replication factor is DC1=3, DC2=3, then configure `nodeFailureTolerance=2`.
3. If Authentication is enabled in Cassandra, then configure the following options in the **[Cassandra]** section under SIP Feature Server application:
  - `username=<cassandra_username>`
  - `password=<Cassandra_password>`
4. If Cassandra SSL encryption is enabled, then perform the following steps:
  - Create a truststore under **<SIP Feature Server installed directory>/etc** and import the public key certificates of the Cassandra nodes.
  - Edit the **<SIP Feature Server installed directory>/launcher.xml** file and set:
    - `com.genesyslab.voicemail.application.cassandraEncryption=true`
    - `javax.net.ssl.trustStore=./etc/truststore` [path of the truststore file]
    - `javax.net.ssl.trustStorePassword=<truststore password>`

---

# Implement the Feature Server GAX Plugin

To install and configure the SIP Feature Server Genesys Administration Extension (GAX) plugin:

1. Log into GAX as an administrator (*GAX IP address:port/gax*).
2. Upload the plugin. Follow the steps in [Uploading Installation Packages](#), using the *Installation Package Upload (template uploaded separately)* method. In the **Installation Package Selection** window, choose these files:
  - Upload a package: `install.zip`
  - Upload an XML template: `fs-gax-plugin.xml`
  - Upload an APD template: `fs-gax-plugin.apd`
3. Install the plugin. Follow the steps in [Deploying Installation Packages](#), selecting the package you uploaded in the previous step.
4. Restart GAX.
5. Ensure that your GAX server and Feature Servers are synchronized within one second of each other.
6. Identify your Feature Server instances. In the GAX application object, under the section **fs-gax-plugin** (or **fs**, but only if **fs** already exists), add an option named `fs_urls`, as needed. Set the option values to a comma-separated list of Feature Server instances, identified by URL. You can use either IP addresses or FQDNs (but not both). Use the format `http://hostname:port/fs`. For example:  
`fs_urls = http://10.10.192.01:8080/fs, http://10.10.192.02:8080/fs`  
or  
`fs_urls = http://fs1:8080/fs, http://fs2:8080/fs`
7. Log into GAX as an administrator. Follow the steps in [Roles](#) to create a new role called, for example, `FS_Access`. Add your users or an access group to the role and assign them the *Access to Genesys SIP Feature Server* privilege.
8. To enable other users to access Feature Server as Administrators, assign the [role privilege](#) *Administrative Access to SIP Feature Server* to individual [users](#) or to [access groups](#).
9. Add your users to the Users access group. See [Access Groups](#).
10. To verify access, log out of GAX and log back in as one of the users (or access group member) that you just provisioned.
11. If you are updating a Feature Server 8.1.200.88 environment that uses the Feature Server dial plan to Feature Server 8.1.201.xx or later, you must [run a migration script](#).

---

# Implement device management

Device management implementation requires installation of the DM GAX plug-in, initial configuration, and setup of required and optional features such as logging and IVR provisioning.

To implement device management:

1. Log into GAX as an administrator (*GAX IP address:port/gax*).
2. Upload the plug-in. Follow the steps in [Uploading Installation Packages](#), using the *Installation Package Upload (template uploaded separately)* method. In the **Installation Package Selection** window, choose these files:
  - Create the package `install.zip` from the contents of the **ip** folder
  - Upload the `install.zip` package
  - Upload the XML template: `dm-gax-plugin.xml`
  - Upload the APD template: `dm-gax-plugin.apd`
3. Install the plug-in. Follow the steps in [Deploying Installation Packages](#), selecting the package you uploaded in the previous step.
4. Restart GAX.
5. Ensure that your GAX server and Feature Servers are synchronized within one second of each other.
6. Create an FQDN, called FQDN2, that includes the IP addresses of all the Feature Server instances (note that this FQDN is different from the one you created in [Configure SIP Feature Server applications](#)). Then configure a DNS server (your choice) to provide cyclic distribution of IP addresses for the FQDN. For example, if 172.24.128.34, 172.24.128.35, and 172.24.128.36 are the IP addresses of three instances, you create the FQDN with all three instances: FQDN2 = `fs1.genesys.com`, which resolves to 172.24.128.34, 172.24.128.35, and 172.24.128.36.
7. Identify your Feature Server instances. In the GAX application object, under the section **[dm-gax-plugin]**, add an option named `fs_urls`, as needed. Set the option values to a comma-separated list of Feature Server instances, identified by URL. Use the format `http://FQDN or FS1 IP address:port/fs`.

**Important:** The order of the first two Features Server instances must be the opposite of the order of the first two Feature Server instances you identified under **[fs-gax-plugin]** or **[fs]**.

For example:

```
[dm-gax-plugin]: fs_urls = http://10.10.192.02:8080/fs, http://10.10.192.01:8080/fs
```

```
[fs-gax-plugin]/[fs]: fs_urls = http://10.10.192.01:8080/fs, http://10.10.192.02:8080/fs
```
8. Grant your users access.
  - In GAX, go to **Configuration > Accounts > Roles** and create a new role.
  - Add users that need access to device management.
  - Under **Assigned Privileges > SIP Device Management**, check **Access to Genesys SIP Device Management**.
9. Set the device management configuration options. In Genesys Administrator, in the Feature Server

---

Application object, under the **[dm]** section:

- set **Active** to true.
- set **fs\_url** to the FQDN and port of the Feature Server that controls the deployment, in the format *FQDN2:port/fs*, where *FQDN2* is the FQDN you created in step 6 above.

For details, see [Configuration options](#).

10. Restart the master Feature Server.

11. To collect the logging data from devices, you must set up a syslog server. Genesys recommends the NXLOG server. To configure your log server:

- Install the NXLOG server, downloadable from <http://sourceforge.net/projects/nxlog-ce/files/>.
- Download the file [nxlog.zip](#) and copy the enclosed **nxlog.conf** file to **nxlog installation directory\conf**, replacing the existing file.
- Create these directories in the nxlog installation directory:
  - **\log\_deposit** (must be a network shared directory)
  - **\log\_backup**

In the case of devices behind SBC/Firewall, the syslog server must be deployed in the same network as the phones and DMs cannot be used for viewing the syslogs. Also, the IP Address that is displayed in the DM UI for the phones may not be accurate.

12. Configure DHCP for phone provisioning. Set the value of the DHCP option (66/160) to:

URL: [http:https://]*FQDN2:port/fs/dm/prov*, where *FQDN2* is the FQDN you created in step 6 above.

- **Note:** For Audiocodes phones that run on firmware 2.0.2.x or below, the DHCP option (66/160) should be set to URL: [http:https://]*FQDN2:port/fs/dm/prov/*.

13. Set up Transport Layer Security (TLS) using the [Genesys Security Deployment Guide](#). To upload the custom-generated server certificates, follow the instructions in [Secure Sockets Layer \(SSL\)](#) (which applies to Feature Server as well as Genesys Web Engagement).

14. For devices behind an SBC, create a trunk with the following options under the **SIP Switch > TServer** section:

- **contact** = *SBC address*
- **oos-options-max-forwards** = 1
- **oosp-transfer-enabled** = true

15. In GA, create a Trunk Group DN named `gcti_provisioning` under the SIP Server switch. Set one configuration option under the **TServer** section:

- **sip-from-pass-through** = true

16. Optionally, set up IVR provisioning, which enables an administrator to use an IVR system to assign an extension to a device.

- Under the same Switch object, create a Trunk DN object (Trunk\_IVR Number). On the Annex tab, in the TServer section, create these options:
  - **contact** = *Resource Manager IP:Port*
  - **prefix** = the prefix of the number to be dialed to access the IVR system. For example, if the IVR number is 888 then prefix can be 8 or 88 or 888.

- 
- Create a resource group of type **gateway** between SIP Server and the GVP Resource Manager.
  - Create a resource group of type **Media control platform** between Media Control Platform and Resource Manager.
  - Create an IVR profile using **Define New IVR Profile**. Add the following options:
    - **service-type** = voicexml
    - **initial-page-url** = [http: https://*Feature Server2 IP address:8080/fs/dm/ivr* (this value uses the first non-master Feature Server IP address)  
*FS(n+1) IP address* is the IP address of the "extra" Feature Server instance that is not included in *FQDN1*.
    - **alternatevoicexml** = [http: https://*FQDN1 or Feature Server1 IP address:8080/fs/dm/ivr* (this value uses either an FQDN or the master Feature Server IP address)  
*FQDN1* is the FQDN you created while **configuring Feature Server applications**, if your environment includes more than two Feature Server instances per SIP switch.
  - Create a DID group and add the IVR profile created in the previous step.
  - Add a DID with the same value as the IVR number that you set under **Administration > SIP Device Management > Settings** in Genesys Administrator Extension.
17. To enable SIP Authentication for a device, add the following option in the **[TServer]** section on the extension DN assigned to the device:
- **authenticate-requests** = Comma-separated list of the following SIP requests:
    - register Enables an authentication procedure on DN registration.
    - invite Authenticates incoming invite requests from the DN.
  - **password** = Any alphanumeric value.
18. You can configure a device that is not listed in the **Supported models**. From the following list, add the options with the appropriate device name to the **[dm]** section of Feature Server application. Use a comma-separated list if you are adding more than one device:
- polycom
  - yealink
  - genesys
  - audiocodes

For example, to add the Polycom SoundStation IP 7000 to the existing support of Polycom SoundStation IP 6000:

```
polycom = SSIP_7000,SSIP_6000
```

---

# Start SIP Feature Server

To start and verify SIP Feature Server:

## Warning

Do not start Feature Server until you have checked that the configuration options are correct in the defined **Cluster** and **Cassandra** sections, specifically **replicationStrategyClassName** and **replicationOptions**. See [Cassandra options](#). Ensure that **Cassandra.yaml** has **endpoint\_snitch: PropertyFileSnitch** if **replicationStrategyClassName** is set to **NetworkTopologyStrategy** and the **resources/cassandra-topology.properties** file has accurate entries describing Feature Server Cassandra ring topology.

1. To run Feature Server in secure (https) mode:
  - Configure your https settings as described in the sections below.
  - In the IVR Profile, set `initial-page-url = https://Feature Server IP address or host name:8443/fs`
2. Use Genesys Administrator, not the command line, to start SIP Feature Server. If you are running more than one Feature Server, start the Master first.
3. In Genesys Administrator, verify that the Feature Server is running.
4. Verify that the GAX interface is running by logging in as the Default administrator (in other words, the Default user in Configuration Server):  
`GAX IP address:port/gax`
5. At this point, only the Default administrator can log into the Feature Server GAX interface. To enable other users to log in as administrators, [assign the Administrator role](#) to them.

## TLS configuration

Minimum Java versions that are required to support TLS in SIP Feature server are:

- Java 6 Update 120 or later
- Java 7 Update 95 or later
- Java 8
- Java 11

By default, SIP Feature Server will use only TLSv1.3. Other protocols will be disabled. Configuration changes with respect to SSL/TLS directly in Java would affect SIP Feature Server and override SIP Feature Server configurations.

In the command section of the **launcher.xml** file, configure:

- Parameter name = **com.genesyslab.voicemail.application.encryption**
- Value = true
- Parameter name = **jdk.tls.client.protocols**
- Value = TLSv1.3

In the **jetty-ssl.context.xml** file, add the following TLS protocol through which SIP Feature Server connections are described.

```
<Set name="IncludeProtocols">
  <Array type="java.lang.String">
    <Item>TLSv1.2</Item>
  </Array>
</Set>
<Set name="ExcludeProtocols">
  <Array type="java.lang.String">
    <Item>TLSv1.1</Item>
    <Item>SSLv3</Item>
  </Array>
</Set>
```

### Important

When TLS is enabled in SIP Feature Server, configure the SIP Feature Server host certificates in the GAX truststore.

## Jetty 9 configuration

This section describes configuring Jetty version 9.

### HTTP configuration

In the command section of the launcher.xml file, configure:

```
parameter name=http_port
default value=jetty.port=8080
```

### HTTPS configuration

This section provides information on HTTPS configuration.

#### Configuration of start.ini

Remove the '#' symbol in start.ini file to enable the HTTPS and SSL parameters listed as follows:

- Enable HTTPS module

```
--module=https
```

- Configure https port  
https.port=8443
- Configure HTTPS idle timeout  
https.timeout=30000
- Enable SSL module  
--module=ssl

## Truststore and keystore configuration paths

Jetty 9 defines main configuration rules for truststore and keystore paths in jetty-ssl-context.xml file. By default, it defines the path as relative to <FS Installation directory>. The default values of Truststore and Keystore path parameters in jetty-ssl-context.xml are as follows:

- <Set name="KeyStorePath"><Property name="jetty.base" default="." /><Property name="jetty.keystore" default="etc/keystore"/></Set>
- <Set name="TrustStorePath"><Property name="jetty.base" default="." /><Property name="jetty.truststore" default="etc/keystore"/></Set>

You can define absolute paths in start.ini by using "jetty.keystore" and "jetty.truststore" variables. In this case, jetty-ssl-context.xml file must be modified as follows:

- <Set name="KeyStorePath"><Property name="jetty.keystore"/></Set>
- <Set name="TrustStorePath"><Property name="jetty.truststore"/></Set>

### Important

The keystore file must not be removed from the <FS Installation directory>/etc/ folder.

Configuring the following keystore and truststore configuration in the start.ini file will override the configuration in the jetty-ssl-context.xml file.

- Setup path to keystore (relative to <FS Installation directory> by default):  
jetty.keystore=etc/keystore
- Setup path truststore (relative to <FS Installation directory> by default):  
jetty.truststore=etc/keystore
- Set the obfuscated passwords for keystore (For more details, see **Generate Obfuscated passwords** topic in this section.):  
jetty.keystore.password=OBF:1vny1zlo1x8e1vnw1vn61x8g1zlu1vn4  
jetty.keymanager.password=OBF:1u2u1wml1z7s1z7a1wn1u2g  
jetty.truststore.password=OBF:1vny1zlo1x8e1vnw1vn61x8g1zlu1vn4

---

## Prevention of Feature Server from POODLE attacks

To prevent Feature Server from POODLE attacks, disable SSLv3 in Feature Server by adding the following to the **Configure** section of the <FS Installation directory>/etc/jetty-ssl.xml file.

```
<Set name="ExcludeProtocols"> <Array type="java.lang.String"><Item>SSLv3</Item>
</Array> </Set>
```

## Generate obfuscated passwords

1. Navigate to <FS Installation directory> in Linux Shell or Windows Command prompt.
  2. Run the following command to run the Jetty's password utility to obfuscate your passwords:  
java -cp lib/jetty-http-xxx.jar:lib/jetty-util-xxx.jar org.eclipse.jetty.util.security.Password your\_Password
- Where -xxx signifies the version of Jetty that you have installed.
  - On Linux, use a colon (:) instead of a semi-colon (;) to separate the two JAR names.

For example:

```
{FS Installation directory}>java -cp lib/jetty-http- 9.2.10.v20150310.jar;lib/
jetty-util- 9.2.10.v20150310.jar org.eclipse.jetty.http.security.Password 123456
123456
OBF:19iy19j019j219j419j619j8
MD5:e10adc3949ba59abbe56e057f20f883e
```

## Configuration of jetty-ssl.xml

In order for the HTTPS connection to select the port as per the configuration, enable the following configuration in the **jetty-ssl.xml** file:

```
<Set name="host"><Property name="jetty.ssl.host" deprecated="jetty.host" /></Set>
<Set name="port"><Property name="jetty.ssl.port" deprecated="ssl.port" default="8443" /></Set>
```

## Jetty 7 configuration

This section describes configuring Jetty version 7.

### HTTP configuration

In the **Options** section of the launcher.xml file, configure:  
parameter name=http\_port  
default value= 8080

### HTTPS configuration

This section provides information on HTTPS configuration.

Remove the '#' symbol in the start.ini file to enable HTTPS in Jetty 7:  
etc/jetty-ssl.xml

The default value of https port in jetty-ssl.xml file:  
<Set name="Port">8443</Set>

## SSL configuration

Jetty 7 defines main configuration rules for SSL parameters in jetty-ssl.xml file. The default values of SSL parameters in jetty-ssl.xml are as follows:

- <Set name="keystore"><SystemProperty name="jetty.home" default="." />/etc/keystore</Set>
- <Set name="password">0BF:1vny1zlo1x8e1vnw1vn61x8g1zlu1vn4</Set>
- <Set name="keyPassword">0BF:1u2u1wml1z7s1z7a1wn1lu2g</Set>
- <Set name="truststore"><SystemProperty name="jetty.home" default="." />/etc/keystore</Set>
- <Set name="trustPassword">0BF:1vny1zlo1x8e1vnw1vn61x8g1zlu1vn4</Set>

## External links for configuring certificates in Jetty

- [Configuring SSL](#)
- [Configuring Jetty SslContextFactory](#)
- [Generating a CSR with keytool](#)
- [Loading Keys and Certificates via PKCS12](#)

---

# Configure SIP Server for Feature Server

To configure the SIP Server application and SIP switch DN's:

1. On the SIP switch that is associated with the SIP Server, create a DN of type VoIP Service (VOIPDN = 9999, for example) to point to the Resource Manager IP address and port, and configure these options in the **Annex > TServer** section:
  - `contact = Resource Manager IP:Port`
  - `service-type = voicemail`
2. To use the Feature Server GAX interface to configure and administer your dial plan: on the SIP switch that is associated with the SIP Server, create a VoIP Service DN named `fs-dialplan` and configure these options:
  - `service-type = extended`  
**Important:** Ensure that you add the final slash character (/) to the end of each of the following URLs. The below-mentioned port indicates the dial plan port of SIP Feature Server.
  - `url = http://FS Node:port/`  
For n+1 High Availability (HA), add the following parameters:
    - `url-1 = http://FS Node2:port/`
    - `url-2 = http://FS Node3:port/`
    - `url-n = http://FS Node_N:port/`
  - **Important:** The dial plan URL of SIP Feature Server must be configured only on a VOIP Service DN that was created on the Switch controlled by the SIP Server that is connected to that particular SIP Feature Server.
3. To use your existing SIP Server dial plan to administer your dial plan, create a DN of type VoIP Service named `standalone-dialplan`. Specify one or more dial plan rules. See the Dial-Plan Rule section in the [Framework 8.1 SIP Server Deployment Guide](#). The following example directs all calls to 2001 to be forwarded to voicemail; 9999 refers to the voicemail VoIP DN configured on the switch. If you use a different number, change the dial plan accordingly.
  - `service-type = dial-plan`
  - `rulename = "2001=>2001;onbusy=9999;ondnd=9999;ontimeout=9999;timeout=1"`
4. In the SIP Server Application object, on the **Options > TServer** tab, configure these options:
  - `dial-plan = fs-dialplan` (the name of the VoIP Service DN created above)
  - `mwi-implicit-notify = true`
  - `subscription-event-allowed = "*"`
5. On the SIP switch that is associated with the SIP Server, create a **DN** of the type **Extension**, and in the **Options > TServer** section under the **Annex** tab, configure these options:
  - `contact = "*"`
  - `gvm_mailbox = mailbox ID`

## Find Me Follow Me

To enable **Find Me Follow Me**:

1. Create a DN of type Trunk Group. In the **TServer** section, add the option:
  - `contact = "Resource Manager IP address:port"`
2. On the SIP switch that is associated with the SIP Server, in the **Options > TServer** section, create a DN of type VoIP Service and configure these options:
  - `contact = Resource Manager IP address:port`
  - `service-type = msml`
3. Copy the file **fmfm-confirmation-prompt-0.wav** from **Feature Server folder\resources\fmfm** on the Feature Server host to **MCP folder/users** on the MCP host.
4. Set the Find Me Follow Me **SIP Server options**.
5. Use the IVR Profile Wizard to create a default IVR Profile. In the section **gvp-general**, add an option **default-application** with any IVR profile as the value.

## Calling Profile for VOIP DN with service type softswitch

You can assign a calling profile to VoIP Service DNs with type softswitch to enable one dial plan for all remote agents. This feature enables agents to use several soft switches and different rules can be applied in each case. To assign a calling profile to VOIP DN with service type softswitch, configure the following options:

- Prerequisite: SIP Server version 8.1.102.59 or above.
1. Create a VoIP DN with the service type softswitch:  
In the TServer section, set (or add) the following options:  
`contact = <remote agent's IP>:< Port>`  
`prefix = <prefix of extension DN for remote agent>`
  2. Create an Extension DN for the remote agent:  
In the TServer section, set (or add) the option:  
`contact = "" (null)`

For more information on this topic, see the Configuring Softswitches section in the [SIP Server Deployment Guide](#).

# Configure voicemail

To configure Feature Server for voicemail and group voicemail:

## Voicemail

1. In Genesys Administrator, in the `rm` section of the Options tab of the Resource Manager (RM) application, set `sip-header-for-dnis` to `request-uri`.
2. Create a resource group of type `gateway` between SIP Server and RM.
3. Create a resource group of type `Media control platform` between MCP and RM.
4. Create an IVR profile using `Define New IVR Profile` and create the following options:
  - `service-type`, value `voicexml`
  - `initial-page-url`, value `[http: https://FQDN1 or FS1 IP address:port/fs FQDN1]` `FQDN1` is the FQDN you created while [configuring Feature Server applications](#), if your environment includes more than two Feature Server instances per SIP switch.
  - `alternatevoicexml`, value `[http: https://FS(n+1) IP address:port/fs FS(n+1) IP address]` `FS(n+1) IP address` is the IP address of the "extra" Feature Server instance that is not included in `FQDN1`.
5. Create a DID group and add the IVR profile created in the step above:
  - Add a DID with the same value as the one set for the VoIP DN option in the previous step. The recommended option and value are `service-type` and `voicexml`.
6. To configure external voicemail deposit, configure a URS strategy that considers the DN/Agent dial-plan settings for business calls that land at a route point. In a TRouteCall to SIP Server, the strategy must respond to the value `full` with `UseDialPlan`.
7. Configure a URS strategy to retrieve an individual or group voicemail. This strategy must respond with a TRouteCall to the P-Alcatel-CSBU header.  
For example, on this routing point the strategy should generate the following TRouteCall:

```
message RequestRouteCall
AttributeThisDN '1519'
AttributeConnID 010a02020db9f03c
AttributeOtherDN 'gcti::voicemail'
AttributeLocation ''
AttributeExtensions [152] 00 04 00 00..
'SIP_HEADERS' 'P-Alcatel-CSBU'
'P-Alcatel-CSBU' 'call_condition=localdirect;categparty=internal;rd=unconditional'
AttributeDNIS '1519'
AttributeRouteType 1 (RouteTypeDefault)
```

---

## Group voicemail

1. To configure group voicemail deposit and retrieval, log into GAX as an administrator (*GAX IP address:port/gax*).
2. Under **Configuration > Configuration Manager**, create a new agent group named Agent\_Group\_1. You can also use a **virtual agent group**, but **only** a virtual agent group based on skill expressions.
3. Create an option:
  - Section: T-Server
  - Name: gvm\_mailbox
  - Value: 2200 (the mailbox number)
4. Add agents to the Agent\_Group\_1 group.
5. Search for any user added to the Agent\_Group\_1 group. Verify that the group mailbox is associated with that user.
6. Create a routing point (for example, 5000).
7. Configure a URS strategy to deposit a group voicemail when none of the agent group members answer the call. This strategy must respond with TRouteCall to gcti::voicemail.
  - Configure the UseDialPlan Extensions attribute with its value set to false.
  - Configure the gvm\_mailbox with its value set to the group mailbox number.

These configurations ensure that SIP Server does not use the SIP Feature Server dial plan and builds an INVITE to the mailbox number provided by the strategy.
8. Load the routing point with the above strategy.
9. Make a call to the route point loaded with this strategy. If no agent in the group answers, after the timeout the call reaches the group mailbox (2200).

## Dial plan processing

For voicemail deposit using the URS strategy, the destination of TRouteCall needs to be provided as a special voicemail forwarding number "gcti::voicemail". If you want to use the regular forwarding number (9999, for example) instead of gcti::voicemail then the number conversion dial plan (dial-plan-rule-1=9999=>gcti::voicemail) does not work by default. The following needs to be configured to support a regular forwarding number.

When a call is routed to any destination using TRouteCall (business calls), then the SIP Server dial plan functionality is not activated. This dial plan restriction is configurable through an AttributeExtension of TRouteCall: UseDialPlan = partial. The possible values of the UseDialPlan extension are: UseDialPlan = false/partial/full/agentid (default=false for internal dial plan functionality).

- If false, the call is routed directly to the target mentioned in the TRouteCall.
- If partial, and the SIP Server dial plan is in effect, SIP Server performs number translation and authorization check on the target processes; if the SIP Feature Server dial plan is in effect, Feature Server performs the number translation and authorization check on the target processes.
- If full, and the SIP Server dial plan is in effect, SIP Server performs number translation, authorization,

and call forwarding; if the SIP Feature Server dial plan is in effect, Feature Server performs number translation, authorization, and call forwarding.

- If `agentid`, Feature Server returns the agent ID of the agent and does not apply digit translation and destination rules. If the agent ID is not available, then Feature Server does not include the **agent-id** field in the XS response, and digits translation and destination rules are not applied.

---

# Configure Regional Voicemail Storage

If any portion of your data center deployment lies in a geographic area where data is required to remain within that area, then you must configure regional restrictions on voicemail storage.

You can now comply with those restrictions by using replication options for regional voicemail keyspaces. With these options, Feature Server can specify that data storage and replication occur in data centers located in a particular region.

## Important

Do not use voicemail while you make these changes, to avoid errors.

### Cassandra-topology.properties file configuration

```
# Cassandra Node IP=Data Center:Rack
#Data Center One
999.999.99.99=us_west:RAC1
999.999.99.98=us_west:RAC2
#Data Center Two
999.999.99.97=us_east:RAC1
999.999.99.96=us_east:RAC2
#Data Center Three
999.999.99.95=eu_east:RAC1
999.999.99.94=eu_east:RAC2
#Data Center Four
999.999.99.93=eu_west:RAC1
999.999.99.92=eu_west:RAC2
# default for unknown nodes
default=us_west:RAC1
```

We'll continue to use `us_east` and `us_west` from the [Adding a New Data Center example](#), (illustrated at left) and add two more for Europe: `eu_east` and `eu_west`. In this configuration:

- Voicemail messages that were processed in the U.S. data centers must be stored in the U.S., and those received in the European data centers must be stored in Europe.

- All voicemail messages must be accessible from any Feature Server nodes in the cluster, regardless of where the messages are physically stored.

This example satisfies the regional voicemail storage requirement by using separate and dedicated Cassandra keyspaces for storing voicemail messages. Your role is to configure the necessary replication options for those special keyspaces. You can follow these instructions, but substitute your installation's specifics (such as key space names) for the names used here.

```
#define keyspace name
regionalKeyspace = us_vm
regionalKeyspace = eu_vm

#number of replication options
regionalReplicationOptions = us_east=2,us_west=2
regionalReplicationOptions = eu_east=2,eu_west=2
```

The example continues. The general (global) keyspace contains these replication options:

```
us_east=2, us_west=2, eu_east=2, eu_west=2
```

Read below how to implement them.

- **regionalKeyspace** defines the keyspace name.  
Add these regional keyspace definitions to the TServer/Cassandra section in all Feature Servers of a region, to create two new regional keyspaces: us\_vm (which will hold voicemails for us\_east and us\_west) and eu\_vm (which will hold voicemails for eu\_east and eu\_west):  
regionalKeyspace = us\_vm  
  
and  
  
regionalKeyspace = eu\_vm
- **regionalReplicationOptions** defines the number of replication options.  
Add these replication option definitions to the TServer/Cassandra section of the master Feature Server and the confSync Feature Server of each region, to configure the number of replication options:  
regionalReplicationOptions = us\_east=2,us\_west=2  
  
and  
  
regionalReplicationOptions = eu\_east=2,eu\_west=2

## Important

You must set **regionalKeyspace** and **regionalReplicationOptions** when all the feature servers of a region are up and running, or before the first start of the feature servers.

- **replicationStrategyClassName** defines the replication strategy. Specify this option in TServer/Cassandra section of Confsync Feature Servers of each region before their first start, to set the replication strategy:  
`replicationStrategyClassName = NetworkTopologyStrategy`

Now, voicemail that is stored at particular location should be accessible from any Feature Server in a Cassandra cluster when all the Feature Servers in a particular region are down; then the voicemails stored in that region are not accessible by any Feature Server.

---

# Configure Cassandra Security

Security configuration for Cassandra enables a secure JMX connection between Feature Server (FS) and its embedded Cassandra.

## Embedded Cassandra JMX Authentication

You can follow this procedure to activate the JMX anonymous authentication and view your FS Cassandra nodes status in the FS UI.

### Important

This feature is not available for versions prior to 8.1.201.82.

1. Edit the launcher.xml file and set the following parameters to true:  
-Dcom.sun.management.jmxremote.authenticate=true
2. Edit the parameter as follows:  
-Dcom.sun.management.jmxremote.password.file=./etc/jmxremote.password
3. Copy: jmxremote.password.template  
from: /jdk\_install\_location/jre/lib/management/  
to: <FS Installation directory>/etc/  
then rename it: jmxremote.password  
  
For OpenJDK 11,  
Copy: jmxremote.password.template  
from: /jdk\_install\_location/jre/conf/management/  
to: <FS Installation directory>/etc/  
then rename it: jmxremote.password
4. Edit the <FS Installation directory>/etc/jmxremote.password file to add the following username:  
fsadmin *yourpassword*  
  
For OpenJDK 11,  
Edit the <FS Installation directory>/etc/jmxremote.password file to add the following username:  
monitorRole *yourpassword*  
controlRole *yourpassword*  
fsadmin *yourpassword*
5. Change the ownership of jmxremote.password to the user you run FS with and change permission to read only.  
For Linux,  
chown fsadmin:fsadmin <FS Installation directory>/etc/jmxremote.password  
chmod 400 <FS Installation directory>/etc/jmxremote.password  
For Windows, see the [Oracle documentation](#).
6. Enable read and write permission to the FS user in:

/jdk\_install\_location/lib/management/jmxremote.access file by adding the following:

```
fsadmin readwrite
```

For OpenJDK 11:

Edit /jdk\_install\_location/conf/management/jmxremote.access file by adding the following:

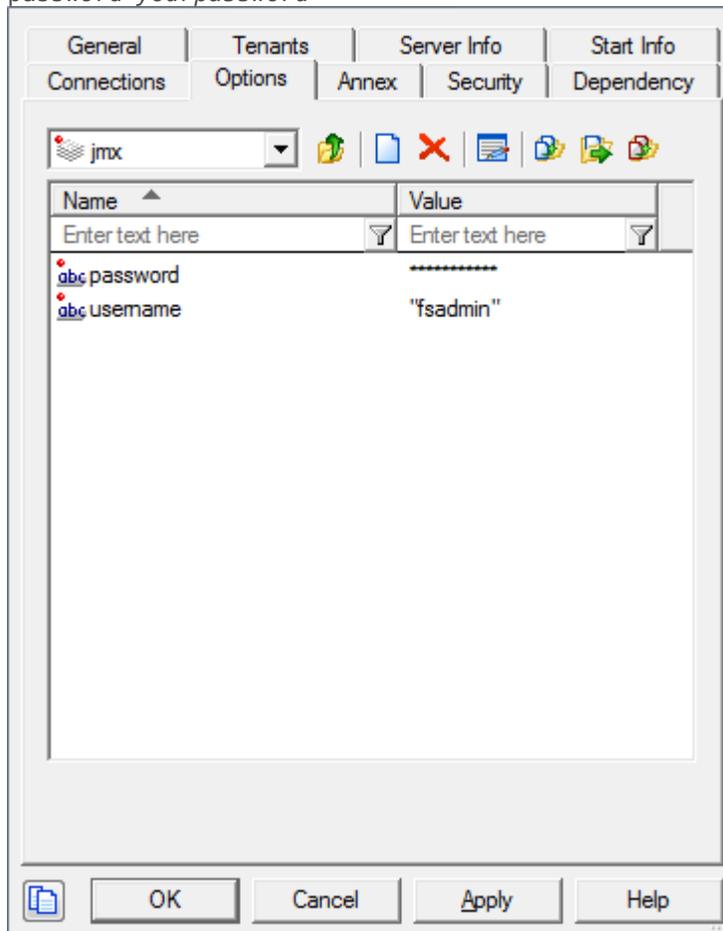
```
fsadmin readwrite
```

7. Edit your FS configuration and create the following options in the **Options** tab:

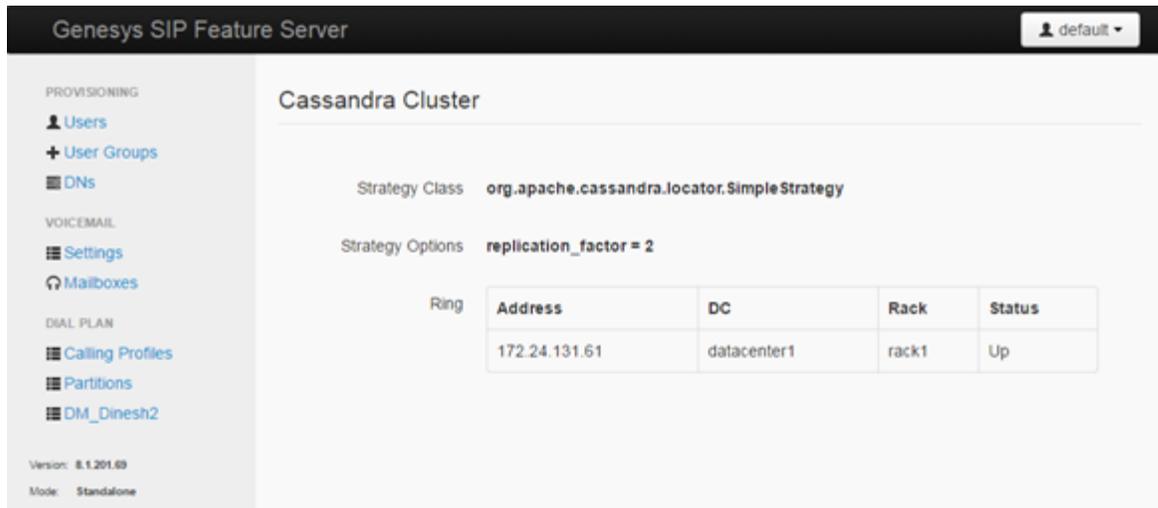
Section jmx

```
username=fsadmin
```

```
password=yourpassword
```



8. Start FS. You can see the status of the Cassandra nodes in:  
[http://<FS\\_HOST>:<PORT>/fs/admin#system/cassandra](http://<FS_HOST>:<PORT>/fs/admin#system/cassandra)



## Cassandra JMX TLS

A Java Management Extensions (JMX) tool manages and monitors Cassandra. The JMX access must be protected to avoid any remote managing on the FS embedded Cassandra.

- To protect JMX access, edit the launcher.xml file and modify the parameters as follows:
  - Dcom.sun.management.jmxremote.ssl.enabled.protocols=TLSv1.3
  - Dcom.sun.management.jmxremote.port=9192
  - Dcom.sun.management.jmxremote.ssl=true
  - Dcom.sun.management.jmxremote.authenticate=true
  - Dcom.sun.management.jmxremote.registry.ssl=true
- Set up Transport Layer Security (TLS). For information on how to create a server certificate, see the [Genesys Security Deployment Guide](#).
- Create a keystore in <FS Installation directory>/etc/ and upload the custom-generated server certificates to the keystore. See the [Oracle documentation](#).
 

**Note:** If FS HTTPS is already enabled with a server certificate, the same keystore and certificate can also be used to secure the embedded Cassandra JMX port.
- Edit and configure the following JVM options in launcher.xml.
  - `javax.net.ssl.trustStore = ./etc/keystore` [path of the trust store file]
  - `javax.net.ssl.trustStorePassword = <trust store password>`
  - `javax.net.ssl.keyStore = ./etc/keystore` [path of the keystore file]
  - `javax.net.ssl.keyStorePassword = <keystore password>`
- Restart the FS to enable a secure JMX connection with the embedded Cassandra.

# Add a New Data Center

If your Cassandra Cluster of Feature Servers *already meets the prerequisites*, then follow the steps to add a new data center to the cluster.

## Prerequisites

### Important

This is not a to-do list; these actions must have been completed before (or when) enabling your Cassandra Cluster. You cannot do them "now".

- All Feature Servers must be installed in Standalone mode with embedded Cassandra cluster.
- The Cassandra cluster must be using Network Topology Strategy.  
The `replicationStrategyClassName` option in the `TServer/Cassandra` section of the master, Feature Server must have been set to `NetworkTopologyStrategy` before the schema was created (before the first/initial start of the 'master' Feature Server).
- The Cassandra cluster must be using the `PropertyFileSnitch` type or the `GossipingPropertyFileSnitch` type of the endpoint snitch.  
You define this option in the `cassandra.yaml` configuration file(s) of all the Feature Servers in the cluster. See [the Cassandra documentation](#) for details.

Do this only when the new data center is created:

- Set one of the new data center's Feature Server nodes to `confSync`.

## Using PropertyFileSnitch

### Cassandra-topology.properties file configuration

```
# Cassandra Node IP=Data Center:Rack
#Data Center One
999.999.99.99=us_west:RAC1
999.999.99.98=us_west:RAC2
#Data Center Two
999.999.99.97=us_east:RAC1
999.999.99.96=us_east:RAC2
#Data Center Three
999.999.99.95=eu_east:RAC1
999.999.99.94=eu_east:RAC2
#Data Center Four
999.999.99.93=eu_west:RAC1
999.999.99.92=eu_west:RAC2
# default for unknown nodes
default=us_west:RAC1
```

1. Deploy new Feature Server nodes for the new data center. [Use this process](#) but ignore the instructions for single data centers.
2. Update Cassandra topology in the `cassandra-topology.properties` file on every Feature Server within the Cassandra Cluster that includes the new nodes.  
**[+] Read instructions:**

In a Cassandra cluster, each Feature Server is a node, and each has an identical `cassandra-topology.properties` file that describes the network topology. When you created the Cassandra Cluster, you created this file and placed a duplicate in the `/resources/` directory of each Feature Server deployment.

You must now update each of these files to enable the new data centers.

The example below began as a single-node configuration. It defined only Data Center One. To configure a new data center, you simply add those same defining lines—modified to contain the correct data. The example below ([modifications in red](#)) now defines four data centers.

```
# Cassandra Node IP=Data Center:Rack
# Data Center One
999.999.99.99=us_west:RAC1
999.999.99.98=us_west:RAC2
<span style="color:red"># Data Center Two
999.999.99.97=us_east:RAC1
999.999.99.96=us_east:RAC2
# Data Center Three
999.999.99.95=eu_east:RAC1
999.999.99.94=eu_east:RAC2
# Data Center Four
999.999.99.93=eu_west:RAC1
```

---

```
999.999.99.92=eu_west:RAC2 </span>  
# default for unknown nodes  
default=us_west:RAC1
```

-To finish your configuration, restart the Feature Servers in existing data center(s).

Read the Cassandra documentation [for additional details](#).

---

3. Restart the existing Feature Server nodes.
4. Change the value of the `endpoint_snitch` type to `PropertyFileSnitch` in newly deployed Feature Servers, then start the new Feature Server nodes [using these instructions](#).
5. Change the Cassandra cluster's keyspace replication options to accommodate the added new data center: specify the replication factor. Be certain that you understand [the Cassandra Keyspace Properties documentation](#) before undertaking this action.
6. The replication options change in the previous step automatically triggers the rebuild process for each new Feature Server/node.  
This process:
  - Runs automatically on all new nodes that you configured in step 2.
  - Streams data for the new nodes from existing nodes.

## Using GossipingPropertyFileSnitch

### cassandra-rackdc-properties file configuration for node1 in new datacenter (DC2)

```
# dc=dc name
# rack=rack name
dc=DC2
rack=RAC1
```

### cassandra-rackdc-properties file configuration for node2 in new datacenter (DC2)

```
# dc=dc name
# rack=rack name
dc=DC2
rack=RAC2
```

1. Deploy new Feature Server nodes for the new data center.
2. Add `cassandra-rackdc.properties` files to the new Feature Server nodes with data center and rack information specific to that node as mentioned in the example below.  
Example: The figure depicts `cassandra-rackdc.properties` files for two Feature Server nodes in a new data center (DC2) to be placed in the Feature Server nodes.
3. Configure `endpoint_snitch` type to `GossipingPropertyFileSnitch` in the `cassandra.yaml` file on the newly deployed Feature Server nodes.
4. Modify the `replicationOptions` option to accommodate the newly added data center in the **[Cassandra]** section of the Feature Server application. Be certain that you understand [the Cassandra Keyspace Properties documentation](#) before undertaking this action.

### Important

When adding a new node to an existing Cassandra cluster:

- Add only one node at a time.
- New nodes must be in the same snitch as the existing nodes snitch. Genesys does not recommend using mixed snitch mode clusters.

# Sample deployments

You can use these sample deployments as a model for your chosen deployment option: single-site, multi-site, or Business Continuity.

## Single-site deployment

### Server configuration

A single-site server could contain the following characteristics:

- SIP Server switch with 2000 extensions.
- Two FS instances as an active-active HA pair, one instance (FS-VM) acting as primary for voicemail, dial plan, and provisioning, and another instance (FS-DM) acting as primary for device management.
- Two IVR profiles, one profile for voicemail and another for device management:
  - IVR profile for voicemail:
    - initial-page-url** = `http://FS-VM IP:port/fs`
    - alternatevoicexml** = `http://FS-DM IP:port/fs`
  - IVR profile for device management:
    - initial-page-url** = `http://FS-DM IP:port/fs/dm/ivr`
    - alternatevoicexml** = `http://FS-VM IP:port /fs/dm/ivr`
- GAX with FS GAX plugin and DM GAX plugin.
  - For example:
    - [dm-gax-plugin]: **fs\_urls** = `http://FS-DM IP:port/fs,http://FS-VM IP:port/fs`
    - [fs-gax-plugin]/[fs]: **fs\_urls** = `http://FS-VM IP:port/fs,http://FS-DM IP:port/fs`
- DNS Server: FQDN (fs1.genesys.com) with the IP address of the two FS instances in DNS Server.
  - For example: fs1.genesys.com, which resolves to IP of FS-DM, IP of FS-VM
- DHCP Server with the provisioning URL as option 66/160. The URL contains the FQDN that belongs to the two FS instances (`http://fs1.genesys.com:port/fs/dm/prov`).
- SIP Server with internal or Feature Server dial-plan
- Both the Feature Server applications include the option **fs\_url** as `http://fs1.genesys.com/8080/fs`.
- In the SIP Switch:
  - a Trunk DN with contact = RM address and prefix = the prefix of the IVR number configured in [Device Management settings](#)
  - a Trunk Group DN with name `gcti_provisioning`
  - For devices behind an SBC, create a trunk with the following options under the **TServer** section:
    - **contact** = *SBC address*
    - **oos-options-max-forwards** = 1

- **oosp-transfer-enabled** = true
- To enable SIP authentication for a device, add the following option in the **[TServer]** section on the extension DN assigned to the device:
  - **authenticate-requests** = register,invite
  - **password** = Any alphanumeric value.

For more information, see how to [Enable SIP authentication for a device](#).

When Feature Server is deployed with co-located/external Cassandra cluster, then configure the nodes option under the Cassandra section of all Feature Server instances as: nodes= Comma-separated IP addresses of [local Cassandra nodes in that data center](#).

### Zero-touch provisioning

1. [Create a device](#) for the IP Phone, assign a profile, and assign DN 1000 to the phone.
2. Connect the IP Phone to the LAN.
  - The phone sends a DHCP a request to get the IP Address.
  - DHCP Server responds to the request with the provisioning URL as option 66/160.
  - The phone sends a configuration file (**MAC.cfg**) request to the provisioning URL:  
http://fs1.genesys.com/fs/dm/prov
  - Feature Server responds with the appropriate configuration file which is already configured.
  - When it receives the configuration file, the phone reboots automatically and is configured with the extension 1000.
  - The phone sends a REGISTER message to SIP Server with DN 1000 and is ready to take calls.

### IVR provisioning

1. Create a [default device management profile](#).
2. Connect the IP Phone to the LAN.
  - The phone sends a DHCP request to get the IP Address.
  - DHCP Server responds IP address with provisioning URL as option 66/160
  - The phone sends the configuration file (**MAC.cfg**) request to the provisioning URL:http://fs1.genesys.com/fs/dm/prov.
  - Feature Server responds with the default configuration file.
  - The phone registers to SIP Server with the special number gcti\_provisioning.
  - When the agent lifts the phone receiver, the phone automatically dials the IVR via SIP Server by using the IVR number.
  - The voice prompt requests the user to assign an extension, and after the extension number 1001 is entered, phone will be notified for getting updated configuration.
  - On phone request, Feature Server provides an updated configuration file and the phone is configured with extension 1001.

- The phone sends a REGISTER message to SIP Server with DN 1001 and is ready to take calls.

## Multi-site deployment

### Server configuration

A multi-site deployment could contain the following server characteristics:

- Two sites, **Site1** and **Site2**.
- Two SIP Switches, one for each site, with 2000 extensions in each switch.
- Four Feature Server instances installed as two active-active pairs, one active-active FS instance per switch:
  - **Site1: FS1-VM, FS1-DM**
  - **Site2: FS2-VM, FS2-DM**
- The DNS Server includes three FQDNs:
  - FQDN1 (fs1.genesys.com) contains the IP addresses of all the FS instances in the DNS Server: FS1-DM, FS1-VM, FS2-VM, and FS2-DM
  - FQDN2 (fs2.genesys.com) contains the IP addresses of the two Site1 FS instances in the DNS Server, in this order: FS1-VM, FS1-DM
  - FQDN3 (fs3.genesys.com) contains the IP addresses of the two Site2 FS instances in the DNS Server, in this order: FS2-DM, FS2-VM
- Three IVR profiles, two for voicemail (Site1 and Site2) and one for device management:
  - IVR profile for voicemail at Site1:  
**initial-page-url** = http://fs2.genesys.com:port/fs  
**alternatevoicexml** = http://fs3.genesys.com:port/fs
  - IVR profile for voicemail at Site2:  
**initial-page-url** = http://fs3.genesys.com:port/fs  
**alternatevoicexml** = http://fs2.genesys.com:port/fs
  - IVR profile for device management (for both sites):  
**initial-page-url** = http://FS1-DM IP:port/fs  
**alternatevoicexml** = http://fs1.genesys.com:port/fs
- GAX with FS GAX plugin and DM GAX plugin, and four FS instances:
  - [dm-gax-plugin]: fs\_urls = http://FS1-DM IP:port/fs,http://FS1-VM IP:port/fs, http://FS2-DM IP:port/fs,http://FS2-VM IP:port/fs
  - [fs-gax-plugin]/[fs]: fs\_urls = http://FS1-VM IP:port/fs,http://FS1-DM IP:port/fs, http://FS2-VM IP:port/fs,http://FS2-DM IP:port/fs
- One DHCP Server with the provisioning URL as option 66/160:  
http://fs1.genesys.com:port/fs/dm/prov

- 
- Two SIP Servers, SIPS1 and SIPS2, use multi-site ISCC configuration.
  - Feature Server application options, in the **dm** section:
    - Site1: option **fs\_url** = `http://fs2.genesys.com:8080/fs` (FQDN2)
    - Site2: option **fs\_url** = `http://fs3.genesys.com:8080/fs` (FQDN3)
  - In each SIP Switch:
    - A Trunk DN with contact = RM address and prefix = the prefix of the IVR number configured in [Device Management settings](#)
    - A Trunk Group DN with name `gcti_provisioning`
    - For devices behind an SBC, create a trunk with the following options under the **TServer** section:
      - **contact** = *SBC address*
      - **oos-options-max-forwards** = 1
      - **oosp-transfer-enabled** = true
    - To enable SIP authentication for a device, add the following option in the **[TServer]** section on the extension DN assigned to the device:
      - **authenticate-requests** = `register,invite`
      - **password** = Any alphanumeric value.

For more information, see how to [Enable SIP authentication for a device](#).

When Feature Server is deployed with co-located/external Cassandra cluster, then configure the nodes option under the Cassandra section of all Feature Server instances as: `nodes= Comma-separated IP addresses of local Cassandra nodes in that data center`.

## Zero-touch provisioning

1. [Create a device](#) for the IP Phone, assign a profile, and assign the DN 1000 to the phone.
2. Connect the IP Phone to the LAN.
  - The phone sends a DHCP request to get the IP Address from the DHCP Server.
  - The DHCP Server responds to the request with the provisioning URL `http://fs1.genesys.com:port/fs/dm/prov` as option 66/160.
  - The phone resolves `fs1.genesys.com` and selects a Feature Server node from Site 1 or Site 2.
  - The phone sends a device configuration file (**MAC.cfg**) request to the selected node with the provisioning URL: `http://fs1.genesys.com/fs/dm/prov`.
  - The selected node sends the provisioned configuration file to the phone.
  - The phone reboots automatically and is configured with the extension 1000 when it starts.
  - The phone sends a REGISTER message to SIPS1 with DN 1000 and is ready to take calls.

## IVR provisioning

1. Create a [default device management profile](#).

2. Complete the necessary configurations for IVR provisioning.
3. Connect the IP Phone to the LAN.
  - The phone sends a DHCP request to get the IP Address from the DHCP Server.
  - The DHCP Server responds to the request with the provisioning URL `http://fs1.genesys.com:port/fs/dm/prov` as option 66/160.
  - The phone resolves `fs1.genesys.com` and selects a Feature Server node from Site 1 or Site 2.
  - The phone sends a configuration file (**MAC.cfg**) request to the selected node with the provisioning URL: `http://fs1.genesys.com/fs/dm/prov`.
  - The selected node sends the default configuration file to the phone.
  - The phone registers to SIP Server with the special number `gcti_provisioning`.
  - When the agent lifts the phone receiver, the phone automatically uses the IVR number to dial IVR through SIP Server.
  - The voice prompt requests the user to assign an extension. When the user enters the extension number 2000, the phone is notified that it needs an updated configuration.
  - On the phone request, Feature Server provides an updated configuration file and the phone is configured with extension 2000.
  - The phone sends a REGISTER message to SIPS2 with DN 2000 and is ready to take calls.

## Business Continuity deployment

### Server configuration

A Business Continuity deployment could contain the following server characteristics:

- Two sites, **Site1** and **Site2**.
- Two SIP Switches, one for each site, with 2000 extensions in each switch.
- All extensions on both switches are synchronized.
- Four Feature Server instances installed as two active-active pairs, one active-active FS instance per switch:
  - **Site1: FS1-VM, FS1-DM**
  - **Site2: FS2-VM, FS2-DM**
- The DNS Server includes three FQDNs:
  - FQDN1 (`fs1.genesys.com`) contains the IP addresses of all the FS instances in the DNS Server: FS1-DM, FS1-VM, FS2-VM, and FS2-DM
  - FQDN2 (`fs2.genesys.com`) contains the IP addresses of the two Site1 FS instances in the DNS Server: FS1-VM, FS1-DM
  - FQDN3 (`fs3.genesys.com`) contains the IP addresses of the two Site2 FS instances in the DNS Server: FS2-VM, FS2-DM
- Business Continuity deployment supports IVR-based device provisioning. The deployment needs three IVR profiles: two for voicemail and one for device management.

- IVR profile for voicemail at Site1:
  - initial-page-url** = http://fs2.genesys.com:port/fs
  - alternatevoicexml** = http://fs3.genesys.com:port/fs
- IVR profile for voicemail at Site2:
  - initial-page-url** = http://fs3.genesys.com:port/fs
  - alternatevoicexml** = http://fs2.genesys.com:port/fs
- IVR profile for device management (for both sites):
  - initial-page-url** = http://fs2.genesys.com:port/fs/dm/ivr
  - alternatevoicexml** = http://fs3.genesys.com:port/fs/dm/ivr
- GAX with FS GAX plugin and DM GAX plugin, and four FS instances:
  - [dm-gax-plugin]: fs\_urls = http://FS1-DM IP:port/fs,http://FS1-VM IP:port/fs, http://FS2-DM IP:port/fs,http://FS2-VM IP:port/fs
  - [fs-gax-plugin]/[fs]: fs\_urls = http://FS1-VM IP:port/fs,http://FS1-DM IP:port/fs, http://FS2-VM IP:port/fs,http://FS2-DM IP:port/fs
- DHCP Server with the provisioning URL as option 66/160. The URL contains the FQDN that belongs to the four FS instances (http://fs1.genesys.com:port/fs/dm/prov).
- Two SIP Servers, SIPS1 and SIPS2, use BC configuration.
- Feature Server application options, in the **dm** section:
  - Site1: option **fs\_url** = http://fs2.genesys.com:8080/fs (FQDN2)
  - Site2: option **fs\_url** = http://fs3.genesys.com:8080/fs (FQDN3)
- In each SIP Switch:
  - A Trunk DN with contact = RM address and prefix = the prefix of the IVR number configured in Device Management settings
  - A Trunk Group DN with name gcti\_provisioning
  - For devices behind an SBC, create a trunk with the following options under the **SIP Switch > TServer** section:
    - **contact** = SBC address
    - **oos-options-max-forwards** = 1
    - **oosp-transfer-enabled** = true
  - To enable SIP authentication for a device, add the following option in the **[TServer]** section on the extension DN assigned to the device:
    - **authenticate-requests** = register,invite
    - **password** = Any alphanumeric value.

For more information, see how to [Enable SIP authentication for a device](#).

- Business Continuity deployment supports IVR-based device provisioning, for DNs that are associated with a Disaster Recovery profile. See [BC Associations](#).

When Feature Server is deployed with co-located/external Cassandra cluster, then configure the

---

nodes option under the Cassandra section of all Feature Server instances as: nodes= Comma-separated IP addresses of **local Cassandra nodes in that data center**.

### Zero-touch provisioning

1. **Create a device** for the IP Phone, assign a profile, and assign DN 1000 to the phone.
2. Connect the IP Phone to the LAN.
  - The phone sends a DHCP request to get the IP Address.
  - DHCP Server responds to the request with provisioning URL as option 66/160.
  - The phone sends the configuration file (**MAC.cfg**) request to the provisioning URL:  
`http://fs1.genesys.com/fs/dm/prov`
  - FS responds with the appropriate configuration file, which is already configured.
  - The phone reboots automatically and is configured with the extension 1000 when it starts.
  - The phone sends a REGISTER message to both SIPS1 and SIPS2 with DN 1000 and is ready to take calls.

### IVR provisioning

1. Create two **Disaster recovery device management profiles** with Preferred and Peer SIP server details for both Site 1 and Site 2.
2. Complete the necessary configurations for IVR provisioning.
3. Connect the IP Phone to the LAN.
  - The phone sends a DHCP request to get the IP Address from the DHCP Server.
  - The DHCP Server responds to the request with the provisioning URL  
`http://fs1.genesys.com:port/fs/dm/prov` as option 66/160.
  - The phone resolves `fs1.genesys.com` and selects a Feature Server node from Site 1 or Site 2.
  - The phone sends a configuration file (**MAC.cfg**) request to the selected node with the provisioning URL: `http://fs1.genesys.com/fs/dm/prov`.
  - The selected node sends the default configuration file to the phone.
  - The phone registers to both Preferred and Peer SIP Servers with the special number `gcti_provisioning`.
  - When the agent lifts the phone receiver, the phone automatically uses the IVR number to dial IVR through Preferred SIP Server.
  - The voice prompt requests the user to assign an extension. When the user enters the extension number 2000, the phone is notified that it needs an updated configuration.
  - On the phone request, Feature Server provides an updated configuration file and the phone is configured with extension 2000.
  - The phone sends a REGISTER message to both SIPS1 and SIPS2 with DN 2000 and is ready to take calls.

## IVR provisioning when the preferred server is down

When the preferred server is down and an agent lifts the phone receiver, the phone automatically uses the IVR number to dial IVR through the Peer SIP Server. The workflow continues by using the Peer SIP server.

# Sample deployments for a co-located/ external Cassandra cluster

You can use these sample deployments as a model for your chosen deployment option: single data center or multi-data center.

## Single data center

A single data center cluster server must contain a minimum of two Cassandra nodes such as Cassandra node 1 and Cassandra node 2.

1. Configure **<Cassandra installed directory>\conf\cassandra.yaml** as follows:
  - cluster\_name: FeatureServerCluster
  - start\_rpc: true
  - listen\_address: Cassandra node IP address/hostname
  - rpc\_address: Cassandra node IP address/hostname
  - seeds: Cassandra node 1 IP address/hostname
  - storage\_port : 7000 (default value)
  - ssl\_storage\_port : 7001 (default value)
  - native\_transport\_port : 9042 (default value)
  - rpc\_port : 9061 (default value)
  - endpoint\_snitch: PropertyFileSnitch
2. Configure **<Cassandra installed directory>\conf\cassandra-topology.properties** as follows:
  - Cassandra node 1 IP address/hostname=DC1:RAC1
  - Cassandra node 2 IP address/hostname=DC1:RAC2

## Multi-data center

Following is a sample deployment for a multi-data centre.

1. A multi data center cluster server must contain a minimum of two data centers:
  - Data center 1 => Cassandra node 1, Cassandra node 2
  - Data center 2 => Cassandra node 3, Cassandra node 4
2. Configure **<Cassandra installed directory>\conf\cassandra.yaml** as follows:
  - `cluster_name`: FeatureServerCluster
  - `start_rpc`: true
  - `listen_address`: Cassandra node IP address/hostname
  - `rpc_address`: Cassandra node IP address/hostname
  - `seeds`: IP addresses/hostnames of Cassandra node 1, Cassandra node3
  - `storage_port` : 7000 (default value)
  - `ssl_storage_port` : 7001 (default value)
  - `native_transport_port` : 9042 (default value)
  - `rpc_port` : 9061 (default value)
  - `endpoint_snitch`: PropertyFileSnitch
3. Configure the **<Cassandra installed directory>\conf\cassandra-topology.properties** as follows:
  - Cassandra node 1 IP address/hostname=DC1:RAC1
  - Cassandra node 2 IP address/hostname=DC1:RAC2
  - Cassandra node 3 IP address/hostname=DC2:RAC1
  - Cassandra node 4 IP address/hostname=DC2:RAC2

## Configuration options

Set these configuration options in Genesys Administrator, in the Options tab of the SIP Feature Server Application object (or the GAX Application object, as noted).

### Cassandra section

Option	Values (default value in bold)	Description
connection-pool-limit	<b>15</b> , or any positive number	Limits the number of connection pools to Cassandra when the rate limiting feature is applied to Voicemail, GAX, and Device Management.  This value must be higher than the value set for <b>request-rate-vm</b> , <b>request-rate-dm</b> and <b>request-rate-gax</b> . When the value is configured as <b>0</b> or not configured, the default value of maximum 15 connections will be allowed. <b>Note:</b> Only SIP Feature Server 8.1.202.40 and later versions support this option.
logFile	<b>cassandra.log</b>	The log file name; applies only to Cassandra logs.
logLevel	<b>error</b> , none, debug, info, warning, critical	The log level; applies only to Cassandra logs.
maxFileSize	<b>20000000</b>	The threshold, in bytes, after which a new log file is created; applies only to dial plan logs.
maxFileCount	<b>2</b>	The number of log files kept at one time; applies only to Cassandra logs.
replicationStrategyClassName	<b>SimpleStrategy</b> , NetworkTopologyStrategy	Specifies the strategy used in distributing data to Cassandra nodes: <ul style="list-style-type: none"> <li>SimpleStrategy uses the value of <b>replicationOptions</b> to determine the number of Cassandra nodes to which the application distributes data. This is used in single data center deployment (Not recommended for production deployment).</li> </ul>

Option	Values (default value in bold)	Description
		<ul style="list-style-type: none"> <li>NetworkTopologyStrategy uses the value of <b>replicationOptions</b> and the values entered in the file <b>resources/cassandra-topology.properties</b> to determine the identity and number of Cassandra nodes to which the application distributes data. This is used in single and multi data center deployment.</li> </ul>
replicationOptions	<p><b>replication_factor=2</b>,                      replication_factor=3,                      replication_factor=4, ...</p> <p>(or)  <i>data center 1 name=number of replication nodes, data center 2 name=number of replication nodes, ...</i></p>	<p>Specifies the number of Cassandra nodes used in the replication scheme, depending on the value of <b>replicationStrategyClassName</b>:</p> <ul style="list-style-type: none"> <li>When <b>replicationStrategyClassName</b> is set to SimpleStrategy, set any of the first group of values to specify the number of Cassandra nodes used in the replication scheme.</li> <li>When <b>replicationStrategyClassName</b> is set to NetworkTopologyStrategy, set any one of the second group of values to specify the data center name and the number of Cassandra nodes used in the replication scheme for each data center.</li> </ul>
readConsistencyLevel and writeConsistencyLevel	<b>CL_ONE</b>	Do not change this value for either option. Determines the minimum number of nodes that data must be written to and read from.
regionalKeyspace	<b>None</b>	<p>Defines the keyspace name for a region. Add this option to create new regional keyspaces. For Example,</p> <p>regionalKeyspace=us_vm (which will hold voicemails for us_east and us_west)                      regionalKeyspace=eu_vm (which will hold voicemails for eu_east and eu_west)</p>
regionalReplicationOptions	<b>None</b>	Defines the number of replication options for a region. Add this option to configure the number of regional replication options. For Example,

Option	Values (default value in bold)	Description
		regionalReplicationOptions = us_east=2, us_west=2 regionalReplicationOptions = eu_east=2, eu_west=2
retry-max-attempts	<b>None</b> , <positive integer value>	Defines the maximum number of times a Feature Server application can retry a failed Cassandra read/write operation.
retry-sleep-ms	<b>None</b> , <positive integer value>	Defines the interval time after which Feature Server should retry a failed Cassandra read/write operation.  To be configured in milliseconds.
nodes	<b>None</b> , Comma-separated Cassandra node IP address or FQDN that are local to its data center	Mandatory if Feature Server is configured with co-located/external Cassandra cluster. Defines the Cassandra nodes where the SIP Feature Server application connects.
keyspace	<b>sipfs</b> , <keyspace name (alpha numeric value)>	Defines the keyspace name to be used by the Feature Server application. By default, sipfs is the Cassandra keyspace name, but it can be changed to any desired alpha numeric name. Applicable only for co-located/external Cassandra cluster.
username	<b>None</b> , <cassandra username>	Configure this option only if authentication is enabled in co-located/external Cassandra. Restart Feature Server for changes to reflect.
password	<b>None</b> , <cassandra password>	Configure this option only if authentication is enabled in co-located/external Cassandra. Restart Feature Server for changes to reflect.
nodeFailureTolerance	<b>None</b> , [replication_factor of its data center - 1].	Specifies that the Feature Server application can process requests until the configured number of Cassandra nodes are down in its local data center.  This option must be configured when the Feature Server application is configured with a replication_factor less than the number of Cassandra nodes in its local data center (For example, if a data center has four nodes, but the replication factor is set to three, then configure this option as replication_factor_value-1. Applicable only for co-located/external Cassandra cluster.

## Cluster section

Option	Values (default value in bold)	Description
master	<b>false</b> , true	<p>When set to <code>true</code>, this option designates this Feature Server as the server that coordinates the synchronization of real-time non-switch data (such as people, agents, and agent groups) from Configuration Server. Synchronization of switch data occurs regardless of this value. The initial import of Tenant data is performed by the master Feature Server only.</p> <p><b>Important:</b> Designate only one server as master.</p>
confsync	<b>false</b> , true	<p>When set to <code>true</code> in a non-master Feature Server, this option enables the server to handle the synchronization of real-time non-switch data with Configuration Server whenever the master server is not operating. Initial import and real-time synchronization of switch data occurs regardless of this value. Value changes take effect immediately.</p>
disable-initial-import	Valid values: <b>false</b> , true	<p>Set to <code>false</code> to enable initial import of tenant data upon startup.</p> <p>To prevent another import, set to <code>true</code>, which disables the initial import trigger after the initial import is completed. Configuring to <code>false</code> will perform an initial import of tenant data and switch data of its switch during the first startup. In rare scenarios, Cassandra data between nodes might become inconsistent. This may appear that the switch data is not available in the Cassandra database. Therefore, SIP Feature Server would re-import all the switch data, resulting in data duplication. To prevent this duplication, configure the option to <code>true</code> after first startup. This disables the initial import trigger in the subsequent starts/restarts.</p>
reimport-on-conf-history-log-error	<b>false</b> , true	<p>If set to <code>true</code>, in the master Feature Server application, then returning from an outage will automatically trigger Reimport.</p>
reimport-on-conf-history-log-error-max-attempts	<b>5</b> , Positive integer value	<p>This limits the number of reimport loops that occur due to permission error on retrieving history log information from the Configuration Server.</p>

Option	Values (default value in bold)	Description
		This option is available for SIP Feature Server 8.1.202.33 and later versions.

## dialplan section

Except for `active`, which indicates whether you are using the Feature Server dial plan or the SIP Server dial plan, these options apply only if you are using the Feature Server dial plan.

Option	Values (default value in bold)	Description
<code>active</code>	<b>true</b> , false	Determines whether Feature Server starts and displays the Feature Server dial plan. Setting the option to false disables the Feature Server dial plan and hides dial plan-related settings in the Feature Server Administration and User web applications. Value change requires restart.
<code>cassandraPoolSize</code>	<b>15</b> and above, or any positive number	Specifies the SIP Feature Server dialplan Cassandra pool size utilized for all Cassandra operations. You must restart SIP Feature Server for the changes to reflect.
<code>default-dn</code>	<b>None</b> , any valid DN	Specifies the DN to which calls are sent when Feature Server is not connected to Cassandra. Applicable only for co-located/external Cassandra cluster.
<code>dialplanEnhancement</code>	<b>false</b> , true	When set to <code>true</code> , this option enables performance monitoring of <b>dialplan</b> such as heartbeat, health check and response code improvements of SIP Feature Server. <div style="border: 1px solid orange; padding: 5px; margin-top: 10px;"> <p><b>Important</b></p> <p>This option is supported only for SIP Feature Server 8.1.202.33 and later versions. To handle the SIP Feature Server error responses and exceptions, SIP Server 8.1.103.79 and later versions must be used.</p> </div>
<code>heartbeatFailInterval</code>	<b>60</b>	The timeout, in seconds, between heartbeat queries from Feature Server to the dial plan when the previous request failed.
<code>heartbeatOkInterval</code>	<b>300</b>	The timeout, in seconds, between

Option	Values (default value in bold)	Description
		heartbeat queries from Feature Server to the dial plan when the previous request succeeded.
logFile	<b>none</b> , <dial plan script working directory>, <another directory>	The log file location on the Feature Server host; applies only to dial plan logs.
logHandler	<b>RotatingFileHandler</b> , DialPlanRotatingFileHandler	When set to DialPlanRotatingFileHandler, the log file is generated in the format <i>logfile_name.YYYYmmdd_HHMMSS_ms.log</i> .  When set to RotatingFileHandler, the log file is generated in the format <i>logfile_name.log</i> . Value change requires a restart.
logLevel	<b>none</b> , debug, info, warning, error, critical	The log level; applies only to dial plan logs.
maxFileSize	<b>1000000</b>	The threshold, in bytes, after which a new log file is created; applies only to dial plan logs.
maxFileCount	<b>2</b>	The number of backup log files kept at one time; applies only to dial plan logs.
port	<b>8800</b> , <other valid http port>	The port specific to the dial plan; must differ from the Feature Server http port.
profileRefreshInterval	<b>300</b>	SIP Feature Server Dialplan refreshes all profiles and partitions into cache after every interval of the timer configured. The values are in seconds. (Valid values can be 1 and above.)  <b>Important</b> This option is supported only for SIP Feature Server 8.1.202.33 and later versions.
tls-version	<b>TLSv1.3</b> , TLSv1.2, TLSv1.1	Dialplan uses this protocol to establish TLS connection. Value change requires restart of Feature Server.

## dm section

Option	Values (default value in bold)	Description
Active	<b>false</b> , true	When set to true, enables Feature Server to manage devices. Restart required.

Option	Values (default value in bold)	Description
fs_url	<i>IP address:port/fs, FQDN:port/fs,</i> <b>none</b>	Specifies the address of the Feature Server that processes Firmware upgrade requests in device management. In an N+1 deployment, the FQDN resolves to the IP address of all the Feature Server instances in that deployment. For example, <i>http://Feature Server FQDN:8080/fs</i> . When this option is not set, Feature Server attempts to detect the IP address automatically and port will be the default Feature Server port. Restart required.
audiocodes/genesys/polycom/yealink	<b>None</b> , comma separated value of the device model names to be supported	Device Management handles and provisions the phone models that are configured in this option, but only if the corresponding firmware version is supported.
UA_Pattern_[audiocodes/genesys/polycom/yealink]	<b>None</b> , parameters with value specific to the phone model	Custom UA header pattern for newly-added phones with a different Header pattern than the default supported pattern.

## dm-gax-plugin section

Set the dm-gax-plugin configuration options in the Options tab of the GAX Application object.

Option	Values (default value in bold)	Description
fs_urls	<i>IP address:port/fs, FQDN:port/fs,</i> <b>none</b>	Specifies the addresses of the Feature Servers that control device management. Specify comma-separated values for multiple Feature Servers. For an N+1 deployment, specify the FQDNs of <i>N</i> servers and the backup server, separated by a comma. For example, <i>http://10.10.192.02:8080/fs, http://10.10.192.01:8080/fs</i> . Restart required.

## fs-gax-plugin section

Set the fs-gax-plugin configuration options in the Options tab of the GAX Application object.

Option	Values (default value in bold)	Description
fs_urls	<i>IP address:port/fs, FQDN:port/fs,</i> <b>none</b>	Specifies the addresses of the Feature Servers that control voicemail management. Specify comma-separated values for multiple Feature Servers. For an N+1 deployment, specify the FQDNs of <i>N</i> servers and the backup server, separated by a comma. For example, http://10.10.192.02:8080/fs, http://10.10.192.01:8080/fs.  Restart required.

## gdpr section

Set this option in the gdpr section on the Options tab of the Master Feature Server application.

Option	Values (default value in bold)	Description
gdpr-directory	An absolute path to the directory with read and write access	The forget-me task will take the input JSON files from this directory. The output log file of the forget-me task will present in this directory.

## Log section

Feature Server uses standard Genesys logging, with the following exceptions to the default values. For details of all logging options, see [Common Configuration Options log section](#).

Option	Values (default value in bold)	Description
internal	<b>Error</b> , Info, Debug	The PSDK logging level; events appear in the same log file.
verbose	<b>trace</b> , all, debug, interaction, standard, none	The verbose logging level.

## Monitoring section

Option	Values (default value in bold)	Description
active	<b>true</b> , false	Determines whether Feature Server collects environment statistics for monitoring. When set to true, the option enables Feature Server

Option	Values (default value in bold)	Description
		environment monitoring to occur even when the monitoring interface is unavailable. To conserve disk space, set the option to false if you are not using environment monitoring.
corrupted-messages-threshold	<b>None</b> , <positive integer value>	Specifies the threshold value of the number of corrupted messages in the mailbox.
logFile	<b>monitoring.log</b> , <i>string.log</i>	The log file name in the Feature Server application directory; applies only to monitoring logs.
logLevel	<b>info</b> , debug, warning, error, critical	The log level; applies only to monitoring logs.
maxFileSize	<b>1000000</b>	The threshold, in bytes, after which a new log file is created; applies only to monitoring logs.
maxFileCount	<b>2</b>	The number of log files kept at one time; applies only to monitoring logs.
monitoring-request-interval	<b>60000</b> (one minute), 10000 (ten seconds) and above	The frequency with which Feature Server requests monitoring information.
prometheus-monitoring	<b>false</b> , true	When set to true, this option enables SIP Feature Server to capture statistics that can be monitored using the Prometheus tool. Value change requires restart of SIP Feature Server.
request-rate-gax	<b>50</b> , zero, or any positive number	The number of requests per second that SIP Feature Server can process from <b>fs-gax-plugin</b> and <b>dm-gax-plugin</b> . When the value is configured as <b>0</b> , there is no rate limit applied. <b>Note:</b> Only SIP Feature Server 8.1.202.40 and later versions support this option. Beginning with SIP Feature Server 8.1.202.40, the default value for this option is 50.
request-rate-vm	<b>50</b> , zero, or any positive number	The number of requests per second that SIP Feature Server can process from Media Control Platform (MCP) during voicemail deposit and access. When the value is configured as <b>0</b> , there is no rate limit applied. <b>Note:</b> Only SIP Feature Server 8.1.202.40 and later versions support this option. Beginning with SIP Feature Server 8.1.202.40, the default value for this option

Option	Values (default value in bold)	Description
		is 50.
request-rate-dm	<b>100</b> , zero, or any positive number	<p>The number of requests per second that SIP Feature Server can process by provisioning requests from IP Phones through Elastic Load Balancing (ELB) and IP phone upgrade requests.</p> <p>When the value is configured as <b>0</b>, there is no rate limit applied.</p> <p><b>Note:</b> Only SIP Feature Server 8.1.202.40 and later versions support this option. Beginning with SIP Feature Server 8.1.202.40, the default value for this option is 100.</p>

## SMTP section

Set these options, used in email notifications of voicemail, in the SMTP section on the Options tab of the SIP Feature Server Application object.

Option	Values (default value in bold)	Description
Host	<i>Server host name</i>	The name of the SMTP host server that you want to use to generate email that notifies users of voicemail.
channel	<b>SMTP</b> , TLS, SSL	The connection protocol that your SMTP host uses. TLS and SSL require a valid username and password.
port	<i>port number</i>	The port number your SMTP server uses, depending on the specified channel.
username	<i>valid username</i>	The username required to log into the designated SMTP server when you have enabled either TLS or SSL security. For example, an e-mail address.
password	<i>valid password</i>	The password for the specified username.
trusted	<b>false</b> , true	When set to <b>true</b> , Feature Server assumes that the specified SMTP server can be trusted, so no server validation through certificates or installation of certificates is necessary.

## TServer section

Set these options, used in Find Me Follow Me, in the TServer section on the Options tab of the SIP Server (**not** SIP Feature Server) Application object.

Option	Values (default value in bold)	Description
fmfm-prompt-file	A valid filename	Specifies the filename of the confirmation prompt. Must match the path and filename in the <b>MCP folder/users</b> folder on the Media Control Platform server host. For example: users/fmfm-confirmation-prompt-0.wav
fmfm-confirmation-digit	<b>0-9</b>	Specifies the digit that a caller must enter for call confirmation. This number must match the number in the recorded fmfm-prompt-file. To use a different digit, you must record a new prompt and place the file in the <b>MCP folder/users</b> folder on the Media Control Platform server host.
fmfm-confirmation-timeout	An integer between 5 and 60; the default value is <b>10</b>	Specifies the timeout value (how long to wait with no input), in seconds. Enter a number that includes playing time of the confirmation prompt and time for the confirmation digit to be entered.  <b>Note:</b> A call is considered abandoned when: the caller hangs up, the entered digit does not match the value of the <b>fmfm-confirmation-digit</b> option, or the call times out with no input at all.
fmfm-trunk-group	A valid Trunk Group DN name	Specifies the Trunk Group DN where events are generated, when each destination leg connects to Media Server. Enter a Trunk Group DN name that represents Media Server. SIP Server uses that DN to play ringback, and for all outbound calls to FMFM destinations.
msml-support	<b>true</b> , false	When set to <b>true</b> , enables the MSML support required to use Find Me Follow Me.

## VoicemailServer section (Application object)

Set these options in the VoicemailServer section on the Options tab of the SIP Feature Server

Application object.

Option	Values (default value in bold)	Description
access-first-prompt-delay-ms	<b>0</b> , zero or any positive integer no greater than 6000 (6000 ms = 6 seconds).	Enables delay in multiples of 600 ms before the first prompt during voicemail access. The delay will be limited to 6000 ms (6 seconds). Defined in milliseconds, the duration value is rounded to the next largest value divisible by 600 ms.  For example, 800 ms is rounded to 1200 ms, 100 ms is rounded to 600 ms, and so on. Overrides the value set at the switch level.
advance-audio-control	<b>true</b> , false	Displays advanced audio controls (forward, rewind, pause, and volume) to the web application user. Overrides the value set at the switch level.
cassandra-counter	<b>true</b> , false	When set to true, the default Cassandra counter is used for calculating voicemail messages count. When set to false, Feature Server uses its own methodology to calculate the voicemail messages count.  <b>Note:</b> This option is supported only for SIP Feature Server 8.1.202.19 and later versions. For upgrading existing deployments, this option should be configured to false only after upgrading all the SIP Feature Server applications to 8.1.202.19 and later versions.
disable	true, <b>false</b>	When set to true, the Voicemail functionality will be disabled for both VUI and TUI. However, when managing mailboxes and messages, the profiles can be still provisioned even when Voicemail is disabled.
disable-general-deposit-prompt	<b>false</b> , true	When set to true, this option disables the prompt, Please leave your message after the tone and press # when you finish, during voicemail deposit.
enable-default-password	<b>false</b> , true	Indicates whether users can use mailbox IDs as default passwords to log on to a mailbox. When set to false, SIP Feature Server allows mailbox IDs to be used as default passwords. When set to true, administrators must assign first-time passwords to all mailboxes. The first-time password can be used to log on to the mailbox once, after which, users must set unique mailbox passwords.  Default passwords are optional only when <b>user-login</b> is set to false. When <b>user-login</b> is set to true, this option does not take effect.
group-optout-use-default	<b>false</b> , true	When set to true, uses the governing value

Option	Values (default value in bold)	Description
		of voicemail-optout-destination as the default group optout prompt. When set to false, does not play any optout prompt if the Group Mailbox does not have an Optout Phone set.
language	<b>&lt;server locale&gt;</b> , de, en-AU, en-GB, en-US, es, es-MX, fr, it, ja, pt-BR, ru, zh-CN, ko	<p>The language used for voicemail prompts; overrides the language set at the switch level, and overridden by the language set for a mailbox:</p> <p><b>Note:</b> The group mailbox language changes based on how it is accessed.</p> <ul style="list-style-type: none"> <li>• When accessed through personal mailbox, the personal mailbox language is used.</li> <li>• When accessed by a non-member or when accessed through a local telephone, the application-level language is used.</li> <li>• When accessed directly, the group mailbox language is used.</li> <li>• When depositing voicemail to group mailbox, the group mailbox language is used.</li> </ul>
locale	<b>en-US</b> , other locale strings	Specifies the default locale for the Telephone User Interface (TUI).
max-concurrent-sessions	Numeric Value	This option restricts the number of concurrent voicemail access sessions for all users. If this option is not configured, there will be no restriction for the concurrent voicemail access, which takes effect for the next voicemail access. This option will be effective only when the <b>user-login</b> option is set to true.
play-disclaimer	<b>false</b> , true	When set to true, enables the system to play a recorded disclaimer; see <a href="#">Provisioning mailboxes</a> . Overrides the value set at the switch level.
play-review-on-deposit	<b>true</b> , false	<p>Specifies whether the Telephone User Interface (TUI) will play the first audio menu prompt: “To confirm press 1, to review press 2, to erase and rerecord press 3.”</p> <p>When set to true, the TUI plays the first audio prompt.</p> <p>When set to false, the TUI omits the first audio prompt and continues as if the caller had pressed choice 1.</p> <p><b>Note:</b> You can set this option here or on the Switch level. If set on both levels, the application level setting</p>

Option	Values (default value in bold)	Description
		takes precedence.
prompt-delay-ms	<b>0</b> , zero or any positive integer in multiples of 600 ms and not greater than 6000 ms (6000 ms = 6 seconds).	Enables delay by introducing silent prompts in multiples of 600 milliseconds before each menu's first prompt during voicemail access. Defined in milliseconds, the duration value is rounded to the next largest value divisible by 600 ms. For example, 800 ms is rounded to 1200 ms, 100 ms is rounded to 600 ms, and so on.  <b>Note:</b> For the <b>home</b> or <b>welcome</b> menu, the value configured in <b>access-first-prompt-delay-ms</b> takes precedence.
security-account-lockout-duration	<b>10</b> , zero or any positive integer	Specifies the time interval, in minutes, that Feature Server waits before unlocking an account that has been locked because of incorrect login entries; the user is locked out after four failed login attempts. A value of 0 (zero) means that only an administrator can unlock the mailbox.
security-max-login-attempt-count	<b>4</b> , any positive integer	Specifies the maximum number of login attempts while accessing an account. The account will be locked after 4 failed login attempts.
security-password-enhanced-validation	<b>false</b> , true	When set to true, forces users to set or reset their passwords to eliminate repeats (333333, for example) and sequences (123456, for example) in mailbox passwords. It also prevents users from using their mailbox number as their password.
security-password-check-internal-call	<b>true</b> , false	Specifies whether the password check request is played when a user dials the voicemail system from an internal phone.
security-password-length-min	<b>4</b> , any positive integer	Specifies the minimum length of the password digits. Increasing this number forces existing users to reset their passwords, if under the new minimum.
skip-optout-prompt	<b>false</b> , true	When set to true, the option disables the prompt, <i>"To interrupt recording and hear more options, press 0 at any time"</i> during the voicemail deposit session.
skip-confirm-ouptout-prompt	<b>false</b> , true	When set to true, the option disables the prompt, <i>"To continue without leaving a message press 1, To leave a message press 2"</i> during the voicemail deposit session.
suspending-state-timeout	<b>-1</b> , any integer greater than 0	Maximum time, in seconds, for which Feature Server will be in suspending state while waiting for ongoing voicemail sessions to be completed.

Option	Values (default value in bold)	Description
		<p>By default (when the option value is set to -1 or no value is configured), Feature Server waits until all the voicemail sessions are completed. This option facilitates a graceful shutdown of the voicemail service during a scheduled maintenance or a software upgrade in Feature Server and ensures the following:</p> <ul style="list-style-type: none"> <li>• Ongoing TUI voicemail sessions are completed before shutdown.</li> <li>• New TUI voicemail sessions are redirected to an alternate Feature Server.</li> <li>• New web UI requests are redirected to an alternate Feature Server.</li> </ul>
time-zone	<b>&lt;server time zone&gt;</b> , any valid time zone (see <a href="http://joda-time.sourceforge.net/timezones.html">http://joda-time.sourceforge.net/timezones.html</a> )	Specifies the default time zone for voicemail messages (applies to the entire application, and overrides any value set at the switch); overridden when the time zone of the user or mailbox is set to a value other than the default.
use-employee-login	<b>None</b> , TController	When set, uses employee ID as agent login code.
user-login	<b>false</b> , true	Indicates whether mailbox validation will be done through a common user password or not.
voice-can-deposit-during-extended-absence	<b>true</b> , false	When set to true, enables the voicemail deposit after the absence greeting is played.
voice-enrollment-enabled	<b>true</b> , false	When set to true, enables the voicemail enrollment.
voice-greeting-extended-max-duration	<b>24</b> , any positive integer	Specifies the length of time, in seconds, allowed for the user to record their extended greeting; after the period ends, a prompt asks the user to confirm their recorded message.
voice-greeting-personal-max-duration	<b>12</b> , any positive integer	Specifies the length of time, in seconds, allowed for the user to record their personal greeting; after the period ends, a prompt asks the user to confirm their recorded message.
voice-message-max-duration	<b>10</b> , any positive integer	Specifies, in seconds, the maximum message length.
voice-message-priority-enabled	<b>false</b> , true	Enables callers to specify whether the voicemail they are leaving is urgent or normal priority. When set to true it plays the message priority selection menu to choose between normal or urgent delivery. When set to false, messages are all sent with normal priority.
voice-mailbox-message-count	<b>10</b> , any positive integer	Specifies the maximum number of voicemail

Option	Values (default value in bold)	Description
		messages per mailbox.
voice-mwi	<b>true</b> , false	If set to false, FS will not generate MWI while handling voicemails.
voicemail-optout-destination	<b>empty (means that the feature is disabled)</b> , any phone number or routing point	When set, enables a caller to transfer out of voicemail to the specified destination at any moment during a call; overrides the voicemail-optout-destination value set at the switch level, but overridden when the Optout Phone field of the Mailboxes settings page has a value. See <a href="#">Provisioning mailboxes</a> .
voicemail-quorum	<b>false</b> , true	Obsolete. The value should always be <b>false</b> .
vxml-access-fetch-timeout	<b>60</b> , any positive integer	This parameter controls the timeout in MCP.

## VoicemailServer section (Switch)

Set these options in the VoicemailServer section on the Options tab of the SIP Switch object, not in the Application object.

Option	Values (default value in bold)	Description
access-first-prompt-delay-ms	<b>0</b> , zero or any positive integer no greater than 6000 (6000 ms = 6 seconds).	Enables delay in multiples of 600 ms before the first prompt during voicemail access. The delay will be limited to 6000 ms (6 seconds). Defined in milliseconds, the duration value is rounded to the next largest value divisible by 600 ms.  For example, 800 ms is rounded to 1200 ms, 100 ms is rounded to 600 ms, and so on. Overridden by the value set at the application level.
advance-audio-control	<b>true</b> , false	Displays advanced audio controls (forward, rewind, pause, and volume) to the web application user. Overridden by the value set at the application level.
language	<b>&lt;server locale&gt;</b> , de, en-AU, en-GB, en-US, es, es-MX, fr, it, ja, pt-BR, ru, zh-CN, ko	The language used for voicemail prompts; overridden by the language set at the application or mailbox level.  <b>Note:</b> The group mailbox language changes based on how it is accessed.  <ul style="list-style-type: none"> <li>When accessed through</li> </ul>

Option	Values (default value in bold)	Description
		<p>personal mailbox, the personal mailbox language is used.</p> <ul style="list-style-type: none"> <li>When accessed by a non-member or when accessed through a local telephone, the application-level language is used.</li> <li>When accessed directly, the group mailbox language is used.</li> <li>When depositing voicemail to group mailbox, the group mailbox language is used.</li> </ul>
play-disclaimer	<b>false</b> , true	When set to <code>true</code> , enables the system to play a recorded disclaimer; see <a href="#">Provisioning mailboxes</a> . Overridden by the value set at the application level, so to enable a disclaimer for a switch, you must also set this option to <code>true</code> at the application level.
play-review-on-deposit	<b>true</b> , false	<p>Specifies whether the Telephone User Interface (TUI) will play the first audio menu prompt: “To confirm press 1, to review press 2, to erase and rerecord press 3.”</p> <p>When set to <code>true</code>, the TUI plays the first audio prompt.</p> <p>When set to <code>false</code>, the TUI omits the first audio prompt and continues as if the caller had pressed choice 1.</p> <p><b>Note:</b> You can set this option here or on the Application level. If set on both levels, the application level setting takes precedence.</p>
time-zone	<b>&lt;server time zone&gt;</b> , any valid time zone (see <a href="http://joda-time.sourceforge.net/timezones.html">http://joda-time.sourceforge.net/timezones.html</a> )	Specifies the default time zone for voicemail messages (applies to the switch); overridden when the time zone of the user, mailbox, or application is set to a value other than the default.
voicemail-optout-destination	<b>empty (means that the feature is disabled)</b> , any phone number or routing point	When set, enables a caller to transfer out of voicemail to the specified destination at any moment during a call; overridden when the voicemail-optout-destination of the application has a value or when the Optout Phone field of the Mailboxes settings page has a value. See <a href="#">Provisioning</a>

Option	Values (default value in bold)	Description
		mailboxes.

## history-log section

Set these options in the history-log section on the Options tab of the Configuration Server application object.

Option	Values (default value in bold)	Description
active	<b>true</b> , false	When Feature Server is disconnected and then reconnected, the changes made in the Configuration Server during the disconnected period are written in the Cassandra database upon reconnection.
expiration	<b>30</b> , 1-30	History log expiration time in days.
max-records	<b>1000</b> , 1-1000	Maximum records stored for retrieving the history log.

## ScheduledTasks section

Set these options in the ScheduledTasks section on the Options tab of the Master Feature Server application.

When SSL and Cassandra authentication is enabled, the following three additional parameters must be appended to the existing values for the **<script name>.cmd** option:

- **-u** *<username>*—The Cassandra username. Default value is "cassandra".
- **--pw** *<password>*—The Cassandra password. Default value is "cassandra".
- **--tls**—Enables SSL connection.

For example, if we consider **update-mailbox-counters** task, **update-mailbox-counters.cmd** must be set as "**refreshInvalidMailboxCounters.py -H localhost -p 9160 -o update-mailbox-counters.log -u cassandra -pw cassandra -tls**".

Option	Values (default value in bold)	Description
update-mailbox-counters.active	<b>false</b> , true	When set to true, this option will enable the task and will schedule the task at the time configured in the update-mailbox-counters.schedule option.

Option	Values (default value in bold)	Description
update-mailbox-counters.cmd	refereshInvalidMailboxCounters.py -H localhost -p 9160 -o update-mailbox-counters.log	Command line for update-mailbox-counters task execution  Command line format: <b>refereshInvalidMailboxCounters.py -H localhost -p 9160 -o &lt;log file name&gt; [-v] [-f all &lt;number&gt;]</b> where: <ul style="list-style-type: none"> <li>• <b>refereshInvalidMailboxCounters.py</b> is the name of the script file located in <b>&lt;FS_installed_path&gt;/python/util</b></li> <li>• <b>-H localhost -p 9160</b> is the Feature Server Cassandra host and port. These parameters may be changed only for external Cassandra deployments.</li> <li>• <b>-o &lt;script log file name&gt;</b> is the script log filename and may include a full path. If the filename is specified, then the default location is <b>&lt;FS_Installed_path&gt;/python.</b></li> <li>• <b>-v</b> enables additional logging (verbose mode).</li> <li>• <b>-f all &lt;number&gt;</b> enables 'fix' mode, which changes invalid counters. If <b>-f</b> parameter is missing, then script generates only the report and no changes are made to the database. The word 'all' can be replaced by an integer, which is the number of changes to be fixed.</li> </ul>
update-mailbox-counters.schedule	0 0 6 ? * SUN	Schedule time for task to run. Should be configured in cron style format.
delete-expired-messages.active	<b>false</b> , true	When set to true, this option will enable and schedule the task at time configured in option delete-expired-messages.schedule.
delete-expired-messages.cmd	removeExpiredMessages.py -H localhost -p 9160 -o delete-expired-messages.log	Command line for delete-expired-messages task execution.  Command line format: <b>removeExpiredMessages.py -H localhost -p 9160 -o &lt;log file name&gt;</b> where:

Option	Values (default value in bold)	Description
		<ul style="list-style-type: none"> <li>• <b>removeExpiredMessages.py</b> is the name of the script file located in <b><i>&lt;FS_installed_path&gt;/python/util</i></b>.</li> <li>• <b>-H localhost -p 9160</b> is the Feature Server/Cassandra host and port. These parameters may be changed only for external Cassandra deployments.</li> <li>• <b>-o &lt;script log filename&gt;</b> is the script log filename and may include a full path. If the filename is specified, then the default location is <b><i>&lt;FS_Installed_path&gt;/python.</i></b></li> </ul> <div style="border: 1px solid orange; padding: 5px; margin-top: 10px;"> <p><b>Important</b></p> <p>For each run, the script will create a new log file <b><i>&lt;script log file name&gt;_&lt;datetime&gt;.log</i></b>.</p> </div>
delete-expired-messages.schedule	0 0 4 ? * SUN	Schedule time for task to run. Should be configured in cron style format.
forget-me.active	<b>false</b> , true	When set to true, this option will enable the task and will schedule the task at time configured in option forget-me.schedule.
forget-me.cmd	forgetMe.py --dbhost <host> --dbport <port>	<p>Command line for forget-me task execution.</p> <p>Command line format: <b>forgetMe.py --dbhost &lt;host&gt; --dbport &lt;port&gt;</b> where:</p> <ul style="list-style-type: none"> <li>• <b>forgetMe.py</b> is the name of the script file located in <b><i>&lt;FS_installed_path&gt;/python/util</i></b>.</li> <li>• <b>--dbhost &lt;host&gt; --dbport &lt;port&gt;</b> is the Feature Server/Cassandra host and port.</li> </ul>
forget-me.schedule	0 0 6 ? * *	Schedule time for task to run. Should be configured in cron-style format.
reimport.active	<b>false</b> , true	When set to true, this option will enable the task and will schedule the task at the time configured in the reimport.schedule option.

---

<b>Option</b>	<b>Values (default value in bold)</b>	<b>Description</b>
reimport.schedule	Time in CRON format	Schedule time for the task to run. Should be configured in CRON style format.

---

# Upgrading SIP Feature Server

Use this procedure to upgrade from one version of SIP Feature Server 8.1.2 to another. Note the following restrictions:

- Transitioning from SIP Voicemail to SIP Feature Server requires a fresh installation.
- You cannot change installation modes (cluster to standalone or vice-versa).

## Important

While upgrading Feature Server, **launcher.xml** will not be updated and the options in **launcher.xml** remain unchanged.

## Important

Beginning with SIP Feature Server version 8.1.203.XX, the **Embedded Cassandra cluster mode** is being deprecated and the feature will be completely removed in future versions. As part of this deprecation, the Embedded Cassandra cluster mode will be removed from the default installation in future versions. If your current deployment model uses the Embedded Cassandra cluster mode for SIP Feature Server, Genesys recommends you to migrate the deployment to other Cassandra modes.

## Upgrading while Feature Server is running (recommended)

To upgrade a running Feature Server environment, stop and upgrade one Feature Server Cassandra cluster instance at a time, **beginning with the master Feature Server**.

1. On the Feature Server master node (which is also the Cassandra seeds node), back up all files in the **etc** folder, which includes the `cassandra.yaml` file.
2. **Stop Feature Server.**
3. Install Feature Server from the installation package. During the upgrade, the installer uses the values provided during a fresh installation.
4. **Start Feature Server.**
5. Repeat steps 2-4 for each Feature Server instance.
6. **Upgrade or install and configure the Feature Server GAX Plug-in.**
7. If you are updating a Feature Server 8.1.200.88 environment that also uses the Feature Server dial plan, you must **run a migration script**.

8. Optionally, create and assign [voicemail profiles](#).

## Upgrading while Feature Server is stopped

To upgrade a running Feature Server environment, upgrade one Feature Server Cassandra cluster instance at a time. Upgrade the master Feature Server last.

1. On the Feature Server master node (which is also the Cassandra seeds node), back up all files in the **etc** folder, which includes the `cassandra.yaml` file. Do **not** upgrade the master node until after you have upgraded all other nodes.
2. On a non-master node, install Feature Server from the installation package. During the upgrade, the installer uses the values provided during a fresh installation.
3. [Start Feature Server](#).
4. Repeat steps 2-3 for each Feature Server instance.
5. [Upgrade or install and configure the Feature Server GAX Plug-in](#).
6. If you are updating a Feature Server 8.1.200.88 environment that also uses the Feature Server dial plan, you must [run a migration script](#).
7. Optionally, create and assign [voicemail profiles](#).

## Restore HTTPS configuration

The following procedure shows how to restore HTTPS configuration of Feature Server after an upgrade. This procedure is applicable only while upgrading Feature Server to version 8.1.201.92 or above.

1. After upgrading to version 8.1.201.93 or above, the Feature Server installation folder contains the following files retrieved from the folder containing the previous versions: **start.ini.bak** and **etc.bak** in both Windows and Linux operating systems. Previously, when upgrading Feature Server overwrote these files rather than retrieving existing files.
2. Compare and copy the difference in the values of **start.ini**, **start.ini.bak** and **etc/jetty-ssl.xml**, **etc.bak/jetty-ssl.xml** and apply the differences in **start.ini** and **etc/jetty-ssl.xml**.
3. Copy **etc.bak/keystore** to **etc/keystore** to restore the configuration.

### Important

After upgrading to Feature Server version 8.1.201.92, the backup of the **etc** folder and the **start.ini** file will be named as **start.ini.backup** (Windows), **start.ini.bak** (Linux), **etc.backup** (Windows), and **etc.bak** (Linux), respectively.

## Post upgrade steps for version 8.1.203 and later

If you have upgraded SIP Feature Server to version 8.1.203 and later and used external Cassandra in your deployment, you can deactivate the Embedded Cassandra module from the deployment. Note that deactivating the Embedded Cassandra module is recommended but it is an optional step.

To deactivate the Embedded Cassandra module:

1. Locate the **start.ini** file in the path: **<FS installation folder>/start.ini**.
2. Open the file with a text editor and remove the line **--module=fs-cass11**.
3. Save the file.
4. Restart SIP Feature Server if it is running.
5. After restart, remove the installation files from the **<FS installation folder>/lib/fs-cass11** folder.

---

# Migrate data from Cassandra database versions

Cassandra versions 2.x and higher do not support backward compatibility with Cassandra versions 1.x. The data migration is required when upgrading Feature Server's Cassandra database backend from versions 1.x to versions 2.x and/or 3.x. Feature Server release 8.1.202.02 includes the following Python scripts for migrating data from Cassandra database version 1.x to versions 2.x and 3.x:

- **copyKeyspaceSchema.py**—Creates a keyspace and its column families in the destination Cassandra cluster.
- **copyKeyspaceColumnFamilies.py**—Copies content of source keyspace column families to the destination keyspace column families.

## Pre-requisites for data migration

The following are the pre-requisites for the data migration from versions 1.x to versions 2.x and/or 3.x.

- Destination Cassandra cluster must be deployed and all the nodes must be up and running.
- In terms of Feature Server deployment, the destination Cassandra cluster must be deployed in external mode.
- The destination Cassandra cluster must not have any Feature Servers assigned to it before the copying of data from the source Cassandra cluster is completed.
- SIP Feature Server must run in the **ReadOnly** mode to ensure proper data copy during migration with running Feature Servers. The **ReadOnly** mode must be turned on before deploying the Python migration scripts. Use the following configuration:  
`[Cassandra]readOnly=true`

## Run Cassandra database migration scripts

The following steps show how to migrate data from Cassandra v1.x to v2.x and v3.x

1. Deploy the Python scripts.
2. Run the Python scripts.
3. Connect Feature Server nodes to migrated Cassandra cluster.

## Deploy the Python scripts

1. Install Python 2.7.5 32-bit version and Pycassa libraries on the destination Cassandra host where the

scripts must be run.

2. The Python scripts **copyKeyspaceSchema.py**, **copyKeyspaceColumnFamilies.py** and the sample json input file, **copyKeyspaceInput.json** are present in the Python utilities folder of Feature Server deployment: *FS installation path/Python/util/*. Copy these script files to a directory on the destination Cassandra host.
3. Navigate to the directory location and run the scripts.

For more details, refer to [Python Scripts](#).

## Run the Python scripts

Following is a sample **copyKeyspaceInput.json** input json file:

```
{
  "sourceHostPort": "FsNode01:9160",
  "sourceHostUserName": "",
  "sourceHostPassword": "",
  "sourceHostTls": "false",
  "destinationHostPort": "CassNode01:9160",
  "destinationHostUserName": "",
  "destinationHostPassword": "",
  "destinationHostTls": "false",
  "replicationStrategyClassName": "NetworkTopologyStrategy",
  "replicationOptions": {"DC1": "2", "DC2": "2"},
  "sourceKeyspace": "sipfs",
  "destinationKeyspace": "sipfs",
  "excludedCFs": [ ],
  "includedCFs": [ ] }

```

## Copy keyspace schema

The following steps show the procedure to copy the keyspace schema:

1. Verify that the input json file has the following parameters:

Parameters	Description	Sample	Mandatory
sourceHostPort	Host and the Thrift port of source Cassandra DB in the URL format: <i>host IP:port</i>	FsNode01:9160	Yes
destinationHostPort	Host and the Thrift port of destination Cassandra database in the URL format: <i>host IP:port</i>	CassNode01:9160	Yes
sourceKeyspace	Name of the source keyspace	sipfs	Yes
destinationKeyspace	Name of the destination keyspace	sipfs	Yes

replicationStrategyClassName	Replication Strategy Class Name	NetworkTopologyStrategy	Yes
replicationOptions	Replication Options for the destination keyspace  Ensure to configure this value according to the <b>cassandra-toplogy.properties</b> file.	{"DC1": "2", "DC2": "2"}	Yes
sourceHostUserName	The username of source Cassandra.	FSadmin	Yes, if authentication is enabled in the source Cassandra Cluster.
sourceHostPassword	The password of source Cassandra.	FSadmin	Yes, if authentication is enabled in the source Cassandra Cluster.
sourceHostTls	Set this option to true when SSL is enabled for the source Cassandra connection.	true	Yes, if SSL is enabled for the source Cassandra.
destinationHostUserName	The username of destination Cassandra.	FSadmin	Yes, if authentication is enabled in the destination Cassandra Cluster.
destinationHostPassword	The password of destination Cassandra.	FSadmin	Yes, if authentication is enabled in the destination Cassandra Cluster.
destinationHostTls	Set this option to true when SSL is enabled for the destination Cassandra connection.	true	Yes, if SSL is enabled for the destination Cassandra.

2. Run the **copyKeyspaceSchema.py** script.

#### Sample command line

```
python ./copyKeyspaceSchema.py -i ./copyKeyspaceInput.json -o
./copyKeyspaceSchema_`date +%Y%m%d-%H:%M`.log
```

### Copy keyspace column families

1. Verify that the input json file has the following parameters:

Parameters	Description	Sample	Mandatory
------------	-------------	--------	-----------

sourceHostPort	Host and the Thrift port of source Cassandra database in the URL format: <i>host IP:port</i>	FsNode01:9160	Yes
destinationHostPort	Host and the Thrift port of destination Cassandra database in the URL format: <i>host IP:port</i>	CassNode01:9160	Yes
sourceKeyspace	Name of the source keyspace	sipfs	Yes
destinationKeyspace	Name of the destination keyspace	sipfs	Yes
excludedCFs	List of comma-separated column family names to be excluded from copying while running the <b>copyKeyspaceColumnFamilies.py</b> script.	message_bytes, device	No
includedCFs	List of comma-separated column family names to be copied while running the <b>copyKeyspaceColumnFamilies.py</b> script.	message_bytes, device	No
sourceHostUserName	The username of source Cassandra.	FSadmin	Yes, if authentication is enabled in the source Cassandra Cluster.
sourceHostPassword	The password of source Cassandra.	FSadmin	Yes, if authentication is enabled in the source Cassandra Cluster.
sourceHostTls	Set this option to <code>true</code> when SSL is enabled for the source Cassandra connection.	<code>true</code>	Yes, if SSL is enabled for the source Cassandra.
destinationHostUserName	The username of destination Cassandra.	FSadmin	Yes, if authentication is enabled in the destination Cassandra Cluster.
destinationHostPassword	The password of destination	FSadmin	Yes, if

	Cassandra.		authentication is enabled in the destination Cassandra Cluster.
destinationHostTls	Set this option to true when SSL is enabled for the destination Cassandra connection.	true	Yes, if SSL is enabled for the destination Cassandra.

If one or more source column families contain huge volumes of data, then run the **copyKeyspaceColumnFamilies.py** script to copy these column families separately from the rest of the source column families. Use the `excludedCFs` and `includedCFs` parameters to exclude or include a specific column family. When the `includedCFs` list is not empty, the `excludedCFs` parameter is ignored and only the column families in the `includedCFs` list are copied.

For example, provide the following json file as the input to the **copyKeyspaceColumnFamilies.py** script to copy the content of all column families except `message_bytes` column family.

```
{
  "sourceHostPort": "FsNode01:9160",
  "sourceHostUserName": "",
  "sourceHostPassword": "",
  "sourceHostTls": "false",
  "destinationHostPort": "CassNode01:9160",
  "destinationHostUserName": "",
  "destinationHostPassword": "",
  "destinationHostTls": "false",
  "replicationStrategyClassName": "NetworkTopologyStrategy",
  "replicationOptions": {"DC1": "2", "DC2": "2"},
  "sourceKeyspace": "sipfs",
  "destinationKeyspace": "sipfs",
  "excludedCFs": [ "message_bytes" ],
  "includedCFs": [ ] }

```

For example, provide the following json file as input to the **copyKeyspaceColumnFamilies.py** script to copy the content of only the `message_bytes` column family.

```
{
  "sourceHostPort": "FsNode01:9160",
  "sourceHostUserName": "",
  "sourceHostPassword": "",
  "sourceHostTls": "false",
  "destinationHostPort": "CassNode01:9160",
  "destinationHostUserName": "",
  "destinationHostPassword": "",
  "destinationHostTls": "false",
  "replicationStrategyClassName": "NetworkTopologyStrategy",
  "replicationOptions": {"DC1": "2", "DC2": "2"},
  "sourceKeyspace": "sipfs",
  "destinationKeyspace": "sipfs",
  "excludedCFs": [],
  "includedCFs": [ "message_bytes" ] }

```

2. Run the **copyKeyspaceColumnFamilies.py** script.

**Sample command line**

```
python ./copyKeyspaceColumnFamilies.py -i ./copyKeyspaceInput.json -o
./copyKeyspaceContent_`date +%y%m%d-%H:%M`.log
```

**Important**

If there are regional keyspaces to be copied, all the keyspaces, the global keyspace and all regional keyspaces must be copied one after the other. To copy all keyspaces, the scripts must be run for each keyspace: the global keyspace and each regional keyspace.

### Connecting Feature Server nodes to migrated Cassandra cluster

The following steps should be performed for every Feature Server node involved:

1. Disable the **ReadOnly** mode in Feature Server. Use the configuration: [Cassandra]readOnly=false
2. Stop Feature Server node.
3. Edit `<FS installation path>\launcher.xml` file and set the property startCassandra to **False**.

```
<parameter name="startCassandra"
displayName="com.genesyslab.common.application.cassandraServer" hidden="true"
mandatory="false">
<description><![CDATA[ Start Cassandra Server]]></description>
<valid-description><![CDATA[]]></valid-description>
<effective-description/>
<format type="string" default="false"/>
<validation>
</validation>
</parameter>
```

4. Update the **[Cassandra]** section of the Feature Server application as shown in the following table:

[Cassandra] section Option	Default Value	Feature Server Application Value	Mandatory
nodes	NA	Configure all the Cassandra nodes IP addresses that belong to the data center where Feature Server is installed.	Yes
nodeFailureTolerance		Replication factor of Feature Server data center is 1.  If the regional keyspace is used, then the least value (keyspace, regional keyspace) replication_factor of its data center is 1.	No

		For example, if the DC1 contains 4 nodes and the replication_factor for the global keyspace is 3 and the regional keyspace is 2, then the value is 1.	
keyspace	sipfs	Name of the 'global' keyspace  This option must have the same value as the keyspace name parameter for the <b>copyKeyspaceSchema.py</b> script when copying the global keyspace.	No
replicationStrategyClassName	NA	This option must have the same value as the replication options parameters for the <b>copyKeyspaceSchema.py</b> script when copying both the global keyspace and the regional keyspace values.	Yes
replicationOptions	NA	This option must have the same value as the replication options parameters for the <b>copyKeyspaceSchema.py</b> script.	Yes
regionalKeyspace	sipfs_<region>	Name of the regional keyspace  This option must have the same value as the replication options parameters for the <b>copyKeyspaceSchema.py</b> script when copying the regional keyspace.	Mandatory if regional keyspace(s) is enabled.
regionalReplicationOptions	NA	This option must have the same value as the replication options parameters for the <b>copyKeyspaceSchema.py</b> script.	Mandatory if regional keyspace(s) is enabled.
username	cassandra	Cassandra Username	Mandatory if authentication is enabled in Cassandra Cluster.
password	cassandra	Cassandra Password	Mandatory if authentication is enabled in Cassandra Cluster.

5. Start Feature Server node.

## Deactivating Embedded Cassandra module for version 8.1.203 and later

If you have upgraded SIP Feature Server to version 8.1.203 and later and used external Cassandra in your deployment, you can deactivate the Embedded Cassandra module from the deployment. Note that deactivating the Embedded Cassandra module is recommended but it is an optional step.

To deactivate the Embedded Cassandra module:

1. Locate the **start.ini** file in the path: **<FS installation folder>/start.ini**.
2. Open the file with a text editor and remove the line **--module=fs-cass11**.
3. Save the file.
4. Restart SIP Feature Server if it is running.
5. After restart, remove the installation files from the **<FS installation folder>/lib/fs-cass11** folder.