API Gateway

Version 7.6.2

14 July 2020

Installation Guide

This documentation describes the following Axway software:

Axway API Gateway 7.6.2

Contents

Preface

This guide describes how to install API Gateway components on all supported platforms.

Note Windows is supported only for a limited set of developer tools, see *Install developer tools on Windows* on page 60. API Gateway and API Manager do not support Windows.

Who should read this guide

The intended audience for this guide is system engineers who are responsible for installing, configuring, and maintaining API Gateway.

Before installing API Gateway you should have an understanding of API Gateway concepts and features. For more information, see the *API Gateway Concepts Guide*.

Others who might find parts of this guide useful include network or systems administrators and other technical or business users.

How to use this guide

This guide should be used in conjunction with the other guides in the API Gateway documentation set.

Before you begin installing API Gateway, review this guide thoroughly. The following is a brief description of the contents of each section:

- *Quick start installation* on page 14 – Enables you to install quickly using standard settings.
- *Plan the deployment* on page 25 – Describes what you should consider when planning for deploying and configuring your system architecture.
- *Prerequisites* on page 16 – Describes the prerequisites for installing, including the system requirements.
- *Install API Gateway* on page 27 – Describes how to perform an installation using the GUI mode or unattended command-line mode.
- *Install API Gateway components* on page 39 – Describes how to install the API Gateway components.
- *Install and configure a metrics database* on page 54 – Describes how to install and configure a metrics database used for monitoring.
- *Install developer tools on Windows* on page 60 – Describes how to install client-side developer tools such as Policy Studio on Windows.

- *Post-installation* on page 61 – Provides instructions on how to check if the installation was successful and describes additional tasks, such as securing API Gateway, that you should perform after installation.
- *Update API Gateway* on page 93 – Describes how to apply service packs or patches to update your API Gateway installation.

API Management documentation set

To find all available documents for this product version:

1. Go to https://docs.axway.com/bundle.
2. In the left pane Filters list, select your product or product version.

Note Customers with active support contracts need to log in to access restricted content.

API Gateway documentation

The API Gateway documentation set includes the following guides:

- *API Gateway Installation Guide*

  Describes how to install API Gateway components on all platforms.
- *API Gateway Upgrade Guide*

  Describes how to upgrade previous API Gateway versions.
- *API Gateway Concepts Guide*

  Provides an overview of the API Gateway components, tools, and architecture.
- *API Gateway Administrator Guide*

  Describes how to configure and manage an API Gateway domain.
- *API Gateway Policy Developer Guide*

  Describes the main API Gateway features and how to configure them using the Policy Studio graphical tool.
- *API Gateway Policy Developer Filter Reference*

  Describes the filters that you can use when developing policies in Policy Studio, and how to configure them.
- *API Gateway DevOps Deployment Guide*

  Describes how to promote and deploy API Gateway configuration between different environments (for example, development, testing, and production).
- *API Gateway OAuth User Guide*

  Describes how to configure API Gateway for OAuth 2.0 and OpenID Connect.
- *API Gateway Developer Guide*

  Describes how to extend, leverage, and customize API Gateway.

- *API Gateway Key Property Store User Guide*

  Describes how to use the Key Property Store (KPS) to configure and manage data referenced from policies running on API Gateway.

- *API Gateway Kerberos Integration Guide*

  Describes how to integrate API Gateway with Kerberos SPNEGO authentication.

- *API Gateway Authentication and Authorization Integration Guide*

  Describes how to integrate API Gateway with Identity Management systems (for example, LDAP servers, CA Siteminder, and so on).

- *API Gateway PassPort Interoperability Guide*

  Describes how to configure API Gateway and Axway PassPort to work together.

- *API Gateway Sentinel Interoperability Guide*

  Describes how to configure API Gateway and Axway Sentinel to work together.

- *API Gateway Validation Authority Interoperability Guide*

  Describes how to configure API Gateway and Axway Validation Authority to work together.

API Manager and API Portal documentation

The API Manager and API Portal documentation set includes the following guides:

- *API Manager User Guide*

  Describes how to use the API management features available separately in API Manager. API Manager is an additional licensable layered product running on API Gateway.

- *API Portal Installation and Upgrade Guide*

  Describes how to install or upgrade API Portal. API Portal is an additional licensable layered product running on API Gateway.

- *API Portal Administrator Guide*

  Describes how to customize and manage API Portal.

- *API Portal User Guide*

  Describes how to use API Portal.

Related documentation

The AMPLIFY API Management solution enables you to create, publish, promote, and manage Application Programming Interfaces (APIs) in a secure and scalable environment. For more information, see the *AMPLIFY API Management Getting Started Guide*.

The following reference documents are also available on the Axway Documentation portal at https://docs.axway.com:

- *Supported Platforms*

  Lists the different operating systems, databases, browsers, and thick client platforms supported by each Axway product.

- *Interoperability Matrix*

  Provides product version and interoperability information for Axway products.

Support services

The Axway Global Support team provides worldwide 24 x 7 support for customers with active support agreements.

Email support@axway.com or visit Axway Support at https://support.axway.com.

See "Get help with API Gateway" in the *API Gateway Administrator Guide* for the information that you should be prepared to provide when you contact Axway Support.

Training services

Axway offers training across the globe, including on-site instructor-led classes and self-paced online learning. For details, go to: http://www.axway.com/support-services/training

Accessibility

Axway strives to create accessible products and documentation for users.

This documentation provides the following accessibility features:

Screen reader support

- Alternative text is provided for images whenever necessary.
- The PDF documents are tagged to provide a logical reading order.

Support for high contrast and accessible use of colors

- The documentation can be used in high-contrast mode.
- There is sufficient contrast between the text and the background color.
- The graphics have the right level of contrast and take into account the way color-blind people perceive colors.

Updates and revisions

This guide includes the following documentation changes.

Changes in version 7.6.2

- Moved the Apache Cassandra administration topics to a new guide. For more information, see *API Gateway Apache Cassandra Administrator Guide*
- Moved the Apache Cassandra installation topic into the API Gateway *Installation* section.
- Updated the prerequisites and installation instructions for Apache Cassandra to state that 2.2.12 is the supported version.
- Updated the prerequisites with additional steps you must complete before installing or running API Gateway if your Linux system has the `/tmp` directory mounted with `noexec`. For more information, see *Additional prerequisites* on page 22.
- Restructured the API Gateway Analytics information and added links to the new *API Gateway Analytics User Guide*.

Changes in version 7.6.1

- The topic on on how perform essential Cassandra operations includes a new section on how to enable Cassandra debug logging.

Changes in version 7.6.0

- Removed the topics on running API Gateway in Docker containers. For detailed information on running API Gateway and API Manager in containers, see the *API Gateway Container Deployment Guide*.
- Removed the installation instructions for Windows for components that are no longer supported on Windows, and added a new section on installing the developer tools still supported on Windows. For details, see *Install developer tools on Windows* on page 60.
- Removed references to API Gateway Appliance and API Gateway Analytics.

Quick start installation 1

This topic describes how to perform a quick start installation of API Gateway on Linux. A quick start installation is a simple, standard installation of API Gateway (for example, for a demonstration or proof of concept).

Note Windows is supported only for a limited set of developer tools, see *Install developer tools on Windows* on page 60. API Gateway and API Manager do not support Windows.

The API Gateway installer provides a default **Standard** installation option, which installs the following API Gateway components:

- API Gateway Server
- QuickStart tutorial
- API Gateway Analytics
- Policy Studio
- Configuration Studio
- Package and deployment tools

The **Standard** option also installs an external Apache Cassandra database, which is used to store API Gateway and API Manager data. For more details, see *Installation options* on page 28.

For more details on API Gateway components and concepts, see the *API Gateway Concepts Guide*.

Before you begin

In preparation for a quick start installation, perform the following tasks:

1. Check that your target system meets the system requirements. For more details, see *Prerequisites* on page 16.
2. Download the installation setup file for your target system.
3. Obtain the necessary license keys from your Axway Account Manager.

Installation

Locate and run the installation setup file. The installer launches in GUI mode by default. Follow the instructions on each window, accepting the default selections at each step. For more information on starting the installer, see *Start installation* on page 28.

When installation is complete, the Cassandra database, API Gateway instance, and Admin Node Manager processes are started, the QuickStart tutorial is launched in a browser window, and the Policy Studio desktop tool is started.

Post-installation

You can use the QuickStart tutorial to invoke some example APIs and to monitor the API Gateway using API Gateway Manager. For more information on using API Gateway Manager, see the *API Gateway Administrator Guide*.

You can use the Policy Studio desktop tool to virtualize APIs and develop policies (for example, to enforce security, compliance, and operational requirements). To begin developing policies in Policy Studio, you must first open or create a new project. For example, follow these steps to create a new project from a running API Gateway instance:

1. When Policy Studio starts up, select **File > New Project**.
2. In the New Project dialog, enter a name for the project and click **Next**.
3. Select **From a running API Gateway instance** and click **Next**.
4. In the Open Connection dialog, select the Admin Node Manager session to connect to, enter the administrator user name and password that you specified during installation and click **OK**.
5. In the Download Options dialog, select the group and the instance to download its configuration.
6. Click **Finish**.

For more information on using Policy Studio, see the *API Gateway Policy Developer Guide*.

Prerequisites 2

This topic describes the prerequisites for installing API Gateway. This includes the system requirements, any platform-specific preparation, required software and licenses, pre-installation tasks, and so on. You must ensure that your target system meets all of the prerequisites before installing API Gateway.

This topic includes the following:

System requirements

This section describes the supported platforms and other system requirements for Axway API Gateway, and specific requirements for API Gateway components.

Note Windows is supported only for a limited set of developer tools, see *Install developer tools on Windows* on page 60. API Gateway and API Manager do not support Windows.

For more details on API Gateway components, see the *API Gateway Concepts Guide*.

Operating systems and hardware

This section describes the operating system requirements for API Gateway.

Platform	Supported versions	Hardware prerequisites
Linux	• CentOS 6.x, 7.x • Oracle Linux 6.x, 7.x • Red Hat Enterprise Linux 6.x, 7.x • SUSE Linux Enterprise Server 11.x, 12.x API Gateway might not run on systems that do not meet these requirements (see **Note** below).	• Supports 64-bit Linux running on 64-bit hardware • Intel Core or AMD Opteron at 2Ghz with Dual Core or faster
Windows (Policy Studio, Configuration Studio, Package and Deployment Tools only)	• Windows 10 • Windows 8.1	• Supports 32-bit Windows on both 32-bit hardware and 64-bit hardware • Intel Core or AMD Opteron at 2Ghz with Dual Core or faster

Note When new Linux kernels and distributions are released, Axway modifies and tests its products for stability and reliability on these platforms. Axway makes every effort to add support for new kernels and distributions in a timely manner.
However, until a kernel or distribution is added to this list, its use with API Gateway is not supported. Axway endeavors to support any generally popular Linux distribution on a release that the vendor still supports.

Disk space and RAM requirements

The disk space and RAM requirements for Linux platforms are:

- Disk space:
  - Minimum 4 GB, 50 GB recommended
- Physical memory (RAM):
  - Minimum 8 GB

The disk space and RAM requirements for the developer tools on Windows platforms are:

- Disk space:
  - Minimum 2 GB
- Physical memory (RAM):
  - Minimum 4 GB

There are also specific requirements for the `/tmp` directory:

- Minimum 500 MB available in the `/tmp` directory and writable permissions on the `/tmp` , `/var/tmp` , and `/usr/tmp` directories.
- `noexec` must not be set on `/tmp`. If `noexec` is set, you must remount `/tmp` with `noexec` disabled or follow the additional steps detailed in */tmp directory mounted with noexec* on page 23.

Databases

This section describes the supported database versions.

Relational databases

API Gateway and API Manager support the following relational databases to store metrics data:

- MySQL Server 5.6, 5.7
- MariaDB 5.5, 10.1
- Microsoft SQL Server 2012, 2014
- Oracle 11.2, 12.1
- IBM DB2 10.5

For more details, see *Install and configure a metrics database* on page 54.

Apache Cassandra

API Gateway and API Manager support Apache Cassandra version 2.2.12 for internal data storage. For more details, see *Install an Apache Cassandra database* on page 36.

Web browsers

API Gateway Manager and other browser-based client components support the following browsers:

- Internet Explorer 11
- Firefox 13.0 or higher
- Safari 5.1.7 or higher
- Google Chrome 19 or higher
- Microsoft Edge (on Windows 10 only)

Thick client platforms

Policy Studio has the following additional requirements on Linux:

- X-Windows environment
- GTK+ 2

Docker containers

API Gateway and API Manager support the following versions of Docker:

- Docker CE version 17.09 or later on CentOS
- Docker EE version 17.06 or later on RHEL

Axway supports Red Hat Enterprise Linux 7 (recommended) and CentOS Linux version 7 as the base image for Docker containers, and supports deployment on any host operating system or cloud provider supported by your Docker version.

Note If you are using API Manager monitoring, you also require a shared file system between your API Gateway instances and Admin Node Manager. This is required for processing of transaction event logs and writing API metrics to a database.

API Gateway elastic topology is supported in Docker deployments only. For more details, see the *API Gateway Container Deployment Guide*.

Specific component requirements

This section describes requirements for specific API Gateway components.

Component	Requirements
Policy Studio	Policy Studio is a thick client and supports the platforms described in *Thick client platforms* on page 19.
API Gateway Manager	API Gateway Manager is a web-based client and supports the web browsers listed in *Web browsers* on page 18.
API Gateway Analytics	The API Gateway Analytics server component has the same operating system and hardware requirements as API Gateway. See *Operating systems and hardware* on page 17. API Gateway Analytics requires a database. For database requirements, see *Databases* on page 18. The browser-based client component supports the same browsers as API Gateway Manager. See *Web browsers* on page 18.

Component	Requirements
API Manager	API Manager is a browser-based client and supports the same browsers as API Gateway Manager. See *Web browsers* on page 18.

Default ports

This section describes the default ports used by API Gateway components.

API Gateway

The default ports used by API Gateway are as follows:

- **Traffic port**: `8080` (between clients and API Gateway)
- **Management port**: `8085` (between API Gateway and Admin Node Manager)

Admin Node Manager

The default port used by the Admin Node Manager for monitoring and management of API Gateway instances is `8090`.

Policy Studio

The default URL address used by the Policy Studio tool to connect to the Admin Node Manager is as follows:

```
https://localhost:8090/api
```

API Gateway Manager

The default URL address used by the API Gateway Manager web console to connect to the Admin Node Manager is as follows:

```
https://localhost:8090/
```

API Manager

The default URL address used by the API Manager web console for API management is as follows:

```
https://localhost:8075/
```

API Gateway Analytics

The default port used by API Gateway Analytics for reporting, monitoring, and management is `8040` . The default URL address used by the API Gateway Analytics web console is as follows:

```
http://localhost:8040/
```

Software and license keys

Axway products are delivered electronically from Axway Support at https://support.axway.com. A welcome email notifies you that your products are ready for download.

When you are ready, perform the following tasks:

1. Check your authorization.
2. Check the hardware and system requirements.
3. Obtain license keys.
4. Download the installation setup file from Axway Support at https://support.axway.com.
5. Install products.

Check your authorization

Verify that you can log in to Axway Support at https://support.axway.com . If you do not have an account, follow the instructions in your welcome email.

Log in to download or access:

- The product installation package
- Product documentation
- Product updates, including patches and service packs
- Product announcements
- The support case center, to open a new case or to track opened cases

You can also access other resources, such as articles in the Knowledge Base, the Axway User Forum, and documentation for all Axway products.

License keys

API Gateway requires the following license keys.

Axway license file

You must have a valid Axway license file to install the following API Gateway components:

- API Gateway Server
- API Gateway Analytics
- API Manager

You can obtain an evaluation trial license to enable you to evaluate the API Gateway features. However, you must have a full license to enable all API Gateway features for use in a non-evaluation environment (for example, development, testing, or production). To obtain an evaluation trial license or a full license, contact your Axway Account Manager.

Note You can install an Admin Node Manager in isolation without an API Gateway license. For more information, see *Install the Admin Node Manager* on page 44.

McAfee license file

You must have a valid McAfee license file to use the **McAfee Anti-Virus** filter.

FIPS-compliant mode license file

You must have a valid Axway FIPS-compliant mode license file to run API Gateway in FIPS-compliant mode.

Multiple installations

API Gateway requires a minimum of two installations for high availability (HA). Make sure that you obtain license keys for all of the API Gateway instances that you are installing.

Additional prerequisites

This section lists additional prerequisites for installing API Gateway.

Executable permission

On Linux, you must ensure that the installation executable has the appropriate permissions in your environment. For example, you can use the `chmod` command to update the file permissions.

/tmp directory mounted with noexec

If your Linux system has the `/tmp` directory mounted with `noexec`, you must complete some additional steps before installing or running API Gateway.

Installation

When installing API Gateway, do not install the QuickStart tutorial:

- When running the installer in GUI mode, you must select the **Custom** setup type and deselect the QuickStart tutorial component. For more information, see *Installation options* on page 28.
- When running the installer in unattended mode, you must use the `--setup_type advanced` option and specify `qstart` to the `--disable-components` option. For more information, see *Unattended installation* on page 33.

You must not install the QuickStart tutorial as this option starts Apache Cassandra, the API Gateway server and the Node Manager when installation completes, and in a system with `/tmp` mounted as `noexec` you must make some changes before starting these components.

Post-installation

After completing the installation and before starting the services:

1. Create a new temporary directory that has `exec` privileges (for example, `/opt/Axway-7.6.2/tmp`).
2. If you installed Cassandra during API Gateway installation, edit the file `CASSANDRA_INSTALL_DIR/conf/cassandra-env.sh` and add the following line:

```
JVM_OPTS="$JVM_OPTS -Djava.io.tmpdir=<TheNewTmpDir>"
```

3. Create or edit the file `VDISTDIR/apigateway/conf/jvm.xml`, and add the following:

```
<ConfigurationFragment>
    <VMArg name="-Djava.io.tmpdir=<TheNewTmpDir>
</ConfigurationFragment>
```

Service packs

Service packs for API Gateway are available from Axway Support at https://support.axway.com. If any service packs are available for API Gateway 7.6.2, download and apply them when the installation completes.

For more information on applying a service pack, see *Update API Gateway* on page 93.

Certificates

API Gateway uses Secure Sockets Layer (SSL) for communications between all processes in a domain (for example, internal management traffic between the Admin Node Manager and API Gateway instances).

Certificates are not required during installation; however, certificates will be required after installation to secure API Gateway domains. For more information on configuring and securing API Gateway domains, see the *API Gateway Administrator Guide*.

Plan the deployment 3

This topic discusses how to plan your deployment. For more information on planning an API Gateway system, and how API Gateway interacts with existing infrastructure, see the *API Gateway Administrator Guide*.

Platforms

For more information on the exact platforms that Axway supports for API Gateway, see *System requirements* on page 16.

Note Windows is supported only for a limited set of developer tools, see *Install developer tools on Windows* on page 60. API Gateway and API Manager do not support Windows.

API Gateway components

Before installing API Gateway you need to consider which components you require. Some components (for example, API Manager or API Gateway Analytics) have additional requirements, such as databases. For more information, see *Specific component requirements* on page 19.

For more information on API Gateway components, see the *API Gateway Concepts Guide*.

Client components

API Gateway includes the Policy Studio developer tool, a thick client that is supported on both Linux and Windows. It also includes several web-based tools (for example, API Gateway Manager and API Gateway Analytics).

For more details on supported thick client platforms and supported web browsers, see *Web browsers* on page 18 and *Thick client platforms* on page 19.

High availability

The following components have specific requirements for high availability (HA):

API Gateway HA

For resilient API Gateway and API Manager HA configuration, a minimum of at least two API Gateway instances is required. For details on configuring API Gateway high availability, see the *API Gateway Administrator Guide*.

Apache Cassandra HA

In addition, the Apache Cassandra database is required to store data for the API Manager component. You can also use Cassandra to store data for API Gateway components such as the Key Property Store, OAuth, and API keys. For Cassandra HA configuration, a minimum of three Cassandra nodes is required. For more details, see "Configure a Cassandra HA cluster" in the *API Gateway Apache Cassandra Administrator Guide*.

Multiple datacenters

For details on how to configure different types of API Gateway and API Manager data in a multi-datacenter environment, see *Configure API Management in multiple datacenters* on page 66.

Connection to other products

API Gateway supports integration with a wide range of Axway products (for example, Axway PassPort) and third-party products (for example, LDAP, JMS, or database providers). The requirements for a deployment of API Gateway with such an integration differs based on the specific product being integrated.

For more details on a particular integration, see the appropriate integration or interoperability guide, available in the Axway Documentation portal at https://docs.axway.com.

For more details on the versions of Axway products that API Gateway 7.6.2 interoperates with, see the following:

- *API Gateway PassPort Interoperability Guide*
- *API Gateway Sentinel Interoperability Guide*
- *API Gateway Validation Authority Interoperability Guide*

Install API Gateway 4

This section describes how to install API Gateway on Linux using the installer.

Note Windows is supported only for a limited set of developer tools, see *Install developer tools on Windows* on page 60. API Gateway and API Manager do not support Windows.

Prerequisites

- You have downloaded the installation setup file for your target operating system from Axway Support at https://support.axway.com.

  The download instructions are in the welcome letter that Axway sent you in an email message.
- You have obtained a valid Axway license file for API Gateway, and optionally API Manager or API Gateway Analytics. Also, if you intend to run API Gateway in FIPS-compliant mode, you have ensured that your license file allows this. You can obtain the required licenses from your Axway account manager.
- You have obtained a valid McAfee license file if you intend to use the **McAfee Anti-Virus** filter.
- You have reviewed the prerequisites and system requirements in *Prerequisites* on page 16 and have ensured that your target system is suitable.

Installation modes

The API Gateway installer has the following installation modes:

- GUI mode
- Unattended command-line mode

The following sections describe how to start the installer in GUI mode and the options that you are presented with when performing a GUI mode installation:

- *Start installation* on page 28
- *Installation options* on page 28

The following section describes how to start the installer in unattended mode and the command-line options for the unattended mode:

- *Unattended installation* on page 33

Start installation

To run the API Gateway installer in the default GUI mode, locate and run the **Linux** setup file. For example:

```
APIGateway_7.6.2_Install_linux-x86-32_BN<n>.run
```

Follow the instructions on each window to complete the installation. For more information on the options available during GUI mode installation, see *Installation options* on page 28.

Note Windows is supported only for a limited set of developer tools, see *Install developer tools on Windows* on page 60. API Gateway and API Manager do not support Windows.

To run the setup in unattended mode, see *Unattended installation* on page 33.

Installation options

When you run the installation setup file it launches in GUI mode by default. The following sections detail the installation options in GUI mode on Linux.

Note Windows is supported only for a limited set of developer tools, see *Install developer tools on Windows* on page 60. API Gateway and API Manager do not support Windows.

Welcome

When you run the setup file in GUI mode, you are presented with an introductory welcome window. Click **Next** to continue with the installation.

License agreement

Read the Axway standard license terms, and click **I accept the agreement** to accept the terms. You cannot proceed with the installation until you make a selection. If you click **I do not accept the agreement**, the installer exits.

Click **Next** to continue.

Select setup type

You can install API Gateway using the following setup types:

Standard	Select this option to install all API Gateway components without API Manager. This includes API Gateway Analytics, the QuickStart tutorial, Apache Cassandra database, package and deployment tools, Policy Studio, and Configuration Studio.
Complete	Select this option to install all API Gateway components with API Manager. This includes API Manager, API Gateway Analytics, the QuickStart tutorial, Apache Cassandra database, package and deployment tools, Policy Studio, and Configuration Studio.
Custom	Select this option to customize which components are installed. You must select this option if you are upgrading from a previous API Gateway version. For more details, see the *API Gateway Upgrade Guide*.

Note The API Tester component is deprecated, and is only installed in a **Custom** setup. For more details, see *Install API Tester* on page 47.

QuickStart tutorial

The **Standard** and **Complete** setup types install the QuickStart tutorial by default, or you can select to install it during the **Custom** setup type. This installs a preconfigured domain and API Gateway instance. If you do not install the QuickStart tutorial, you must configure a domain and API Gateway instance when the installation is complete. For more details, see *Initial configuration* on page 62.

Click **Next** to continue.

Select components

This window is only displayed during an **Custom** installation.

Select the components to be installed, and deselect those that are not to be installed. The following components are selected by default:

- API Gateway Server
- Admin Node Manager
- Policy Studio desktop tool

Click **Next** to continue.

Specify installation directory

Enter a location or click the browse button to specify the directory where the API Gateway components are to be installed, for example:

```
/opt/Axway-7.6.2
```

Click **Next** to continue.

Specify license file

Enter the location or click the browse button to specify a valid Axway license file. For more details, see *Software and license keys* on page 21.

Note API Gateway, API Gateway Analytics, and API Manager each require a valid Axway license file. If you have separate license files for each of these components, specify the API Gateway license at this step, and you will be prompted for the API Gateway Analytics and API Manager license files at a later step. Alternatively, you can specify a single license file that covers all licensed components.

Cassandra configuration

If you selected to install an Apache Cassandra database, configure the following settings:

- **Installation Directory**:
  Enter the directory in which to install the Cassandra server (for example, `/opt/db/cassandra`).

  Caution Do not install Apache Cassandra in the same directory as the API Gateway components to avoid errors during the Cassandra upgrade.

- **JRE Location**:
  Enter the directory of the Oracle Java Runtime Environment used by Cassandra. The default value is the location of the Oracle JRE provided by API Gateway (for example, `INSTALL_DIR/apigateway/Linux.x86_64/jre/bin`). If you have installed a separate Oracle JRE for Cassandra, enter its location instead.

  For details of the Cassandra JRE requirements and recommendations, see *Cassandra prerequisites* on page 36.

Set the administrator credentials for the Admin Node Manager

It is important to secure your API Gateway system to protect it from internal and external threats. This window enables you to set the administrator user name and password used to log in to Policy Studio and API Gateway Manager. These administrator credentials are also used by `managedomain` when connecting to an Admin Node Manager.

Select **Change the default user name and password** to set the user name and password for the administrator account, and enter a user name and password. This option is selected by default to ensure that you set your own administrator user name and password. To use a default administrator user name and password, you must deselect this option. The default credentials are available from your Axway account manager.

Caution You must ensure that you remember these credentials or you will not be able to log in to Policy Studio or API Gateway Manager. This is especially important when planning to install Policy Studio on Windows later because you do not have the option to set the credentials then.

Click **Next** to continue.

Specify QuickStart Node Manager details

This window is only displayed if you selected to install the QuickStart tutorial.

Configure the following settings for the Node Manager:

- **Host Name or IP Address**:
  Select a host address from the list (defaults to the installation host name).
- **Local Management Port**:
  Enter the local port used to manage the Node Manager. Defaults to `8090`.

Click **Next** to continue.

Specify QuickStart server details

This window is only displayed if you selected to install the QuickStart tutorial.

Configure the following settings:

- **Local Management Port**:
  Enter the local port that the Node Manager uses to manage the API Gateway instance. Defaults to `8085`.
- **External Traffic Port**:
  Enter the port that the API Gateway uses for message traffic from external clients. Defaults to `8080`.

Click **Next** to continue.

Set the administrator credentials for API Manager

It is important to secure your API Manager system to protect it from internal and external threats. This window enables you to set the API administrator user name and password used to log in to the API Manager web console.

Select **Change the default user name and password** to set the user name and password for the API administrator account, and enter a user name and password. This option is selected by default to ensure that you set your own API administrator user name and password. To use a default API administrator user name and password, you must deselect this option. The default credentials are available from your Axway account manager.

Caution Ensure that you remember these credentials or you will not be able to log in to API Manager.

Click **Next** to continue.

Set the administrator credentials for API Gateway Analytics

It is important to secure your API Gateway Analytics system to protect it from internal and external threats. This window enables you to set the administrator user name and password used to log in to the API Gateway Analytics web console.

Select **Change the default user name and password** to set the user name and password for the administrator account, and enter a user name and password. This option is selected by default to ensure that you set your own administrator user name and password. To use a default administrator user name and password, you must deselect this option. The default credentials are available from your Axway account manager.

Caution You must ensure that you remember these credentials or you will not be able to log in to API Gateway Analytics.

Click **Next** to continue.

Installation summary

The installer displays a summary of the components that will be installed on your system.

Review the information and click **Next** to begin installing.

Installing

A progress window is displayed showing the progress of the installation. When the installation is complete, click **Next** to continue.

Installation complete

A window is displayed to indicate that the installation is complete. If you selected to install Policy Studio you can select the option to **Launch Axway Policy Studio**.

The URL of the Admin Node Manager is displayed (for example, `https://127.0.0.1:8090`). You can go to this URL in your browser to access the API Gateway Manager tools.

Click **Finish** to complete the installation. Policy Studio is launched if you selected that option. If you selected to install the QuickStart tutorial, it is also launched in a browser window.

Unattended installation

This topic explains how to run the API Gateway installer in unattended mode on Linux and Windows. It also describes each of the available command options.

Note Windows is supported only for a limited set of developer tools, see *Install developer tools on Windows* on page 60. API Gateway and API Manager do not support Windows.

Run the installer in unattended mode

You can run the API Gateway installer in unattended mode on the command line. Perform the following steps:

1. Change to the directory where the setup file is located.
2. Run the setup file with the `--mode unattended` option.

Standard setup without API Manager

The following example shows how to install all available API Gateway components (excluding API Manager on Linux) in unattended mode:

Linux

```
./APIGateway_7.6.2_Install_linux-x86-32_BN<n>.run --mode unattended
--setup_type standard
--licenseFilePath my_license.lic
--analyticsLicenseFilePath my_analytics_license.lic
--prefix /opt/Axway-7.6.2
--cassandraInstalldir opt/db/cassandra
--cassandraJDK opt/jre
--startCassandra 1
```

Windows

```
APIGateway_7.6.2_Client_Tools_Install_win-x86-32_BN<n>.exe --mode unattended
--prefix C:\Axway-7.6.2

--analyticsLicenseFilePath my_analytics_license.lic
```

The components are installed in the background, in the directory specified by the `--prefix` option. On Windows, the installed components are Policy Studio, Configuration Studio, and Package and Deployment Tools only.

Complete setup with API Manager

The following example shows how to install all API Gateway components on Linux, including API Manager, in unattended mode:

```
./APIGateway_7.6.2_Install_linux-x86-32_BN<n>.run --mode unattended
--setup_type complete
--licenseFilePath my_license.lic
--analyticsLicenseFilePath my_analytics_license.lic
--apimgmtLicenseFilePath my_mgmt_license.lic
--prefix /opt/Axway-7.6.2
--cassandraInstalldir /opt/db/cassandra
--cassandraJDK /opt/jre
--startCassandra 1
```

The components are installed in the background, in the directory specified by the `--prefix` option.

Custom setup

The topics on installing each API Gateway component show how to use the `--setup_type advanced` option to install a custom setup in unattended mode. For example, see *Install the API Gateway server* on page 39.

Unattended mode options

For a description of all the available command-line options and their default settings, run the setup file with the `--help` option. This outputs the help text in a separate console. For example:

Linux

```
./APIGateway_7.6.2_Install_linux-x86-32_BN<n>.run --help
```

Windows

```
APIGateway_7.6.2_Client_Tools_Install_win-x86-32_BN<n>.exe --help
```

The following table summarizes some of the more common options:

Option	Description
`--help`	Display available options and default settings.
`--mode`	Specify an installation mode.

Option	Description
`--setup_type`	Specify a setup type (`standard`, `complete`, or `advanced`) (on Linux only).
`--enable-components`	Specify a comma-separated list of components to enable.
`--disable-components`	Specify a comma-separated list of components to disable.
`--prefix`	Specify an installation directory.
`--licenseFilePath`	Specify the path to a license file (on Linux only).
`--analyticsLicenseFilePath`	Specify the path to an API Gateway Analytics license file.
`--apimgmtLicenseFilePath`	Specify the path to an API Manager license file (on Linux only).
`--unattendedmodeui`	Specify different levels of user interaction when installing on a UNIX/Linux system with X-Windows or on Windows.
`--cassandraInstalldir`	Specify the Apache Cassandra installation directory, for example, `opt/db/cassandra` (on Linux only).
`--cassandraJDK`	Specify the location of your Oracle Java Runtime Environment for Apache Cassandra. The default value is: • `INSTALL_DIR/apigateway/Linux.x86_64/jre/bin` **Note** OpenJDK is not supported.
`--startCassandra`	Specify whether the Apache Cassandra server starts after the installer completes (on Linux only). Set to `1` to start Cassandra after installation, or set to `0` if you do not want Cassandra to start.
`--optionfile`	Specify options in a properties file. For more information on option files, go to: http://installbuilder.bitrock.com/docs/installbuilder-userguide.html

Install an Apache Cassandra database

Apache Cassandra is required to store data for API Manager (for example, API catalog, quotas, and client registry) or API Gateway client registry (API key and OAuth). In addition, Cassandra is optional to store data for the following API Gateway components:

- Custom KPS table definitions and data
- OAuth token stores

Note You must ensure that Cassandra is installed and running to use API Manager or API Gateway client registry.

Supported Cassandra versions

API Gateway supports Apache Cassandra version 2.2.12. For more details on Apache Cassandra, see http://cassandra.apache.org/.

For details on upgrading your Cassandra version, see "Upgrade Apache Cassandra" in the *API Gateway Upgrade Guide*.

Upgrade from earlier API Gateway versions

API Gateway version 7.5.3 and later include the Datastax Cassandra client, which uses a default port of `9042` to communicate with Cassandra over the Cassandra native protocol. Earlier API Gateway versions included the Hector Cassandra client, which used a default port of `9160` to communicate with Cassandra over the Apache Thrift protocol.

In API Gateway version 7.5.1 or later, Cassandra runs externally to the API Gateway process. In earlier API Gateway versions, Cassandra was embedded in the API Gateway process.

For details on upgrading from an earlier API Gateway version, see the *API Gateway Upgrade Guide*.

Cassandra prerequisites

This section describes Cassandra-specific prerequisites in addition to the general API Gateway *Prerequisites* on page 16.

Production environment requirements

API Gateway supports the following in production:

- **Operating systems**:
  - All supported Linux platforms (see *System requirements* on page 16)

- **Cassandra**:
  - Cassandra version 2.2.12
  - 64-bit Oracle JRE version 8 (OpenJDK is not supported)

For details on requirements for high availability, see "Configure a Cassandra HA cluster" in the *API Gateway Apache Cassandra Administrator Guide*.

JRE requirements and recommendations

The default API Gateway installation includes a 64-bit Oracle JRE (`apigateway/Linux.x86_64/jre/bin`). You can configure Cassandra to use the API Gateway JRE (for example, in a demo environment), but it is recommended that you install a separate Oracle JRE for use with Cassandra. When using a separate JRE, use the same version (or at least the same major version) as the API Gateway uses.

JCE policies for Cassandra TLS/SSL

If client TLS/SSL will be enabled for Cassandra, you must install the Java Cryptographic Extension (JCE) policies for your Oracle JRE. For example, you can download the Oracle Java 8 JCE policies from:

http://www.oracle.com/technetwork/java/javase/downloads/jce8-download-2133166.html

Install Apache Cassandra

Note Apache Cassandra 2.2.12 is installed by default in an API Gateway Standard or Complete setup. For more details, see *Installation options* on page 28.

Install Cassandra in GUI mode

In GUI mode, to install Apache Cassandra only, use the steps described in *Installation options* on page 28 with the following selections:

- **Setup Type**: Select **Custom**.
- **Select Components**: Select **Cassandra**.
- **Cassandra configuration**: Enter your Cassandra **Installation Directory** and your **JRE Location**. For more details, see *Cassandra configuration* on page 30.

Install Cassandra in unattended mode

To install Apache Cassandra using the API Gateway installer in unattended mode, follow the steps described in *Unattended installation* on page 33.

The following command is an example of how to install Apache Cassandra in unattended mode on Linux:

```
./APIGateway_7.6.2_Install_linux-x86-32_BN<n>.run --mode unattended
--setup_type advanced
--enable-components cassandra
--disable-components apigateway,qstart,policystudio,analytics,
configurationstudio,apitester,apimgmt,packagedeploytools
--cassandraInstalldir /opt/db/cassandra
--cassandraJDK /opt/jre
--startCassandra 0
```

Keep Cassandra installation after API Gateway is uninstalled

To keep your Cassandra installation after API Gateway is uninstalled, you must ensure that you first install Cassandra only. For example, perform the following steps:

1. Run the API Gateway installer, and select Cassandra only.
2. Run the API Gateway installer, and select API Gateway components to install.

If API Gateway is uninstalled, Cassandra remains installed.

For more details on Apache Cassandra, see the following:

- http://cassandra.apache.org/
- http://docs.datastax.com/en/cassandra/2.2/

Install API Gateway components

5

This topic describes how to install each API Gateway component separately. The API Gateway installer enables you to perform the following:

- *Install the API Gateway server* on page 39
- *Install the QuickStart tutorial* on page 41
- *Install the Admin Node Manager* on page 44
- *Install Policy Studio* on page 45
- *Install API Tester* on page 47
- *Install Configuration Studio* on page 46
- *Install API Manager* on page 48
- *Install the Package and Deploy tools* on page 50
- *Install API Gateway Analytics* on page 51

For more details on API Gateway components and concepts, see the *API Gateway Concepts Guide*.

Install the API Gateway server

The API Gateway server is the main runtime environment consisting of an API Gateway instance and a Node Manager. This topic describes how to install API Gateway on Linux.

Note Windows is supported only for a limited set of developer tools, see *Install developer tools on Windows* on page 60. API Gateway and API Manager do not support Windows.

For more details on API Gateway components and concepts, see the *API Gateway Concepts Guide*.

Prerequisites

Ensure that all of the prerequisites detailed in *Prerequisites* on page 16 are met.

Axway license file

You must have a valid Axway license file to install the API Gateway server. Also, if you intend to run API Gateway in FIPS-compliant mode, ensure that your license file allows this. To obtain an evaluation trial license or a full license, contact your Axway Account Manager.

Note If you are using Apache Cassandra, before starting API Gateway, you must first ensure that Cassandra is installed and running. For more details, see *Install an Apache Cassandra database* on page 36.

Install the API Gateway server

To install the API Gateway server in GUI mode, perform an installation following the steps described in *Installation options* on page 28, using the following selections:

- Select the **Custom** setup type.
- Select to install the API Gateway server component.

To install the API Gateway server in unattended mode, follow the steps described in *Unattended installation* on page 33.

The following example shows how to install the API Gateway server component in unattended mode:

```
./APIGateway_7.6.2_Install_linux-x86-32_BN<n>.run --mode unattended
--setup_type advanced
--enable-components apigateway
--disable-components nodemanager,qstart,policystudio,analytics,
apitester,configurationstudio,apimgmt,cassandra,packagedeploytools
--licenseFilePath mylicense.lic
```

Before you start API Gateway

Note Before you can start API Gateway, you must first use the `managedomain` script to create a new domain that includes an API Gateway instance. If you installed the QuickStart tutorial, a sample API Gateway domain is automatically configured in your installation. Otherwise, you must first create a new domain. For more details, see the *API Gateway Administrator Guide*.

If you installed the QuickStart tutorial, the QuickStart server and Admin Node Manager start automatically. Otherwise, you must start them manually.

If you installed Apache Cassandra, before starting API Gateway, you must first ensure that your Apache Cassandra server is running. For more details, see *Install an Apache Cassandra database* on page 36.

Start API Gateway

To start API Gateway manually, follow these steps:

1. Open a command prompt in the following directory:

```
INSTALL_DIR/apigateway/posix/bin
```

2. Run the `startinstance` command, for example:

```
startinstance -n "Server1" -g "Group1"
```

Note You must ensure that the `startinstance` has execute permissions.

3. To manage and monitor API Gateway, you must ensure that the Admin Node Manager is running. Use the `nodemanager` command to start the Admin Node Manager from the same directory.
4. To launch API Gateway Manager, enter the following address in your browser:

```
https://HOST:8090/
```

`HOST` refers to the host name or IP address of the machine on which API Gateway is running (for example, `https://localhost:8090/`).

5. Enter the administrator user name and password. This is the administrator user name and password you entered during installation.

Note You can encrypt all sensitive API Gateway configuration data with an encryption passphrase. For example, you can specify this passphrase in your API Gateway configuration file, or on the command line when the API Gateway is starting up. For more details, see the *API Gateway Administrator Guide*.

Start as a service

You can also run API Gateway instances and Node Managers as services. For more information, see *Set up services* on page 64.

Install the QuickStart tutorial

The API Gateway QuickStart tutorial is available on Linux. It demonstrates the main API Gateway features and tools, and enables you to invoke some example APIs and to monitor API Gateway using API Gateway Manager.

Tip The QuickStart tutorial is automatically installed as part of a default **Standard** or **Complete** setup. For more details, For more details, see *Installation options* on page 28.

Prerequisites

Ensure that all of the prerequisites detailed in *Prerequisites* on page 16 are met.

Install the QuickStart tutorial

Note The QuickStart tutorial is dependent on the API Gateway Server. You cannot install the QuickStart tutorial without the API Gateway Server.

To install the API Gateway Server and the QuickStart tutorial in GUI mode, perform an installation following the steps described in *Installation options* on page 28, using the following selections:

- Select the **Custom** setup type.
- Select to install the API Gateway Server, Admin Node Manager, and QuickStart tutorial components.

To install the API Gateway Server, Admin Node Manager, and QuickStart tutorial in unattended mode, follow the steps described in *Unattended installation* on page 33.

The following example shows how to install the API Gateway Server component and the QuickStart tutorial in unattended mode:

```
APIGateway_7.6.2_Install_linux-x86-32_BN20200714.run --mode unattended
--setup_type advanced
--enable-components apigateway,nodemanager,qstart
--disable-components,policystudio,apitester,
configurationstudio,apimgmt,cassandra,packagedeploytools
--licenseFilePath mylicense.lic
```

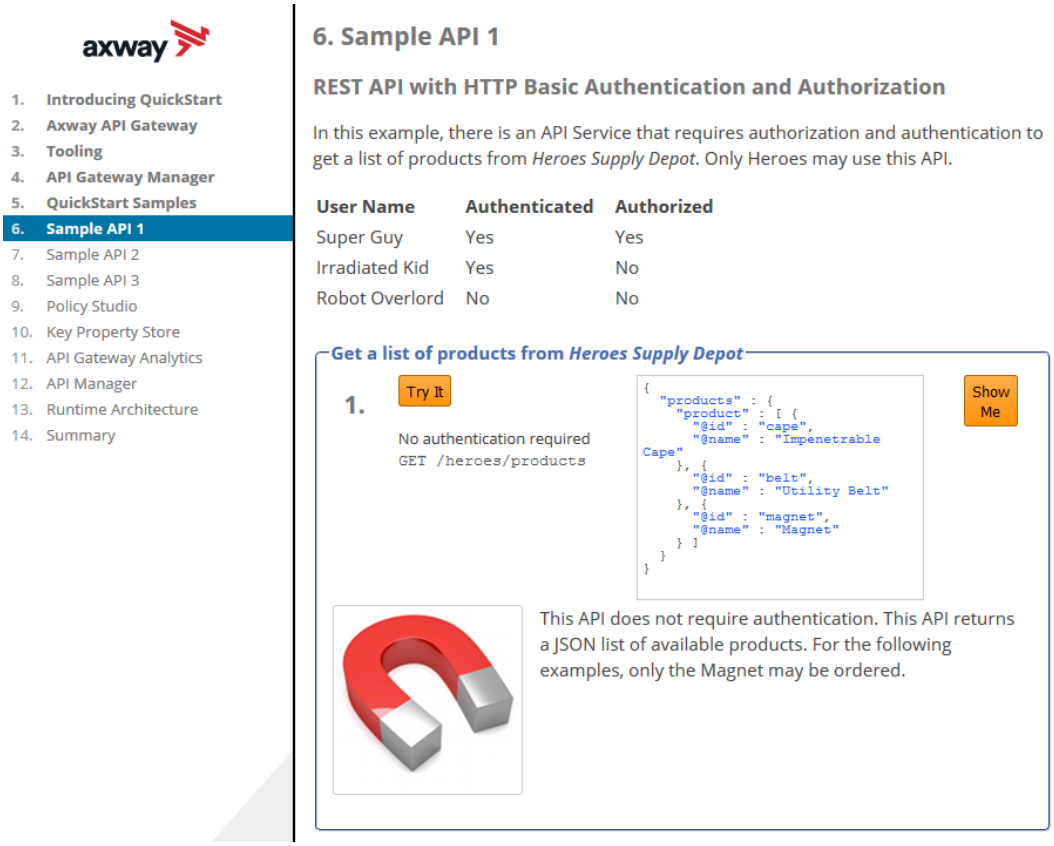
QuickStart domain configuration

When the QuickStart tutorial is installed, a sample API Gateway domain is automatically configured in your installation. This includes a `QuickStart Server` API Gateway instance that runs in a `QuickStart Group` group. The QuickStart server and Admin Node Manager start automatically when installation is complete.

Start the QuickStart tutorial

The QuickStart tutorial launches automatically in your browser when installation is complete. Follow the instructions in your browser to perform the steps in the tutorial.

For example, the following screen shows invoking a sample API in the tutorial:



You can click the **Try it** button to invoke the sample API. This displays a JSON list of available products. You can click the **Show Me** button to view the traffic monitored by API Gateway in API Gateway Manager.

Restart the QuickStart tutorial

At any point, if you need to restart the QuickStart tutorial, perform the following steps:

1. Open a command prompt in the following directory:

   ```
   INSTALL_DIR/apigateway/posix/bin
   ```

2. Run the `startinstance` command, for example:

   ```
   startinstance -n "QuickStart Server" -g "QuickStart Group"
   ```

   Note You must ensure that the `startinstance` has execute permissions.

3. To manage and monitor the API Gateway, you must ensure that the Admin Node Manager is running. Use the `nodemanager` command to start the Admin Node Manager from the same directory.

4. To launch API Gateway Manager, enter the following address in your browser:

```
https://127.0.0.1:8090/
```

5. Enter the administrator user name and password. This is the administrator user name and password you entered during installation.
6. To launch the QuickStart tutorial, enter the following address in your browser:

```
http://127.0.0.1:8080/quickstart/index.html?mgr=8090
```

Install the Admin Node Manager

You can install an Admin Node Manager component on Linux in isolation without an API Gateway license. For more details on API Gateway components and concepts, see the *API Gateway Concepts Guide*.

Prerequisites

Ensure that all of the prerequisites detailed in *Prerequisites* on page 16 are met.

Install the Admin Node Manager

To install the Admin Node Manager in GUI mode, perform an installation following the steps described in *Installation options* on page 28, using the following selections:

- Select the **Custom** setup type.
- Select to install the Admin Node Manager component.

To install the Admin Node Manager in unattended mode, follow the steps described in *Unattended installation* on page 33.

The following example shows how to install the Admin Node Manager component in unattended mode:

```
./APIGateway_7.6.2_Install_linux-x86-32_BN<n>.run --mode unattended
--setup_type advanced
--enable-components apigateway
--disable-components qstart,policystudio,configurationstudio,analytics,
apitester,apimgmt,cassandra,packagedeploytools
```

Start the Admin Node Manager

For more information on starting the Admin Node Manager, see *Start API Gateway* on page 40.

Install Policy Studio

Policy Studio is a graphical IDE that enables developers to virtualize APIs and develop policies to enforce security, compliance, and operational requirements. You can install Policy Studio on both Linux and Windows.

Note Windows is supported only for a limited set of developer tools, see *Install developer tools on Windows* on page 60. API Gateway and API Manager do not support Windows.

For more details on API Gateway components and concepts, see the *API Gateway Concepts Guide*.

Prerequisites

Ensure that all of the prerequisites detailed in *Prerequisites* on page 16 are met.

Install Policy Studio

To install Policy Studio in GUI mode, perform an installation following the steps described in *Installation options* on page 28, using the following selections:

- Select the **Custom** setup type. This screen is omitted on Windows.
- Select to install the Policy Studio component.

To install Policy Studio in unattended mode, follow the steps described in *Unattended installation* on page 33.

The following example shows how to install the Policy Studio component in unattended mode on Linux:

```
./APIGateway_7.6.2_Install_linux-x86-32-BN<n>.run --mode unattended
--setup_type advanced
--enable-components policystudio
--disable-components nodemanager,apigateway,qstart,apitester,
analytics,configurationstudio,apimgmt,cassandra,packagedeploytools
```

Start Policy Studio

Note Before starting Policy Studio, ensure both the Admin Node Manager and the API Gateway instance are running. For more details, see *Start API Gateway* on page 40.

If you did not select to launch Policy Studio after installation, perform the following steps:

1. Open a command prompt.
2. Change to your Policy Studio installation directory (for example, `INSTALL_DIR/policystudio`).

3. Run `policystudio`.
4. When Policy Studio starts up, select **File > New Project**.
5. In the New Project dialog, enter a name for the project and click **Next**.
6. Select **From a running API Gateway instance** and click **Next**.

Tip You can also create configuration projects from `.fed` files or from existing configurations. For more information, see the *API Gateway Policy Developer Guide*.

7. In the Open Connection dialog, select the Admin Node Manager session to connect to, enter the administrator user name and password you specified when you installed API Gateway, and click **OK**.
8. In the Download Options dialog, select a group and an API Gateway instance to download its configuration.
9. If a passphrase has been set, enter it in the **Passphrase** field, and click **Finish**. Alternatively, if no passphrase has been set, click **Finish**. For more details on setting a passphrase, see the *API Gateway Administrator Guide*.

Install Configuration Studio

Configuration Studio is a graphical tool that enables administrators to configure environment-specific properties to deploy APIs and policies in non-development environments. You can install Configuration Studio on bith Linux and Windows.

Note Windows is supported only for a limited set of developer tools, see *Install developer tools on Windows* on page 60. API Gateway and API Manager do not support Windows.

For more details on using Configuration Studio, see the *API Gateway DevOps Deployment Guide*.

Prerequisites

Ensure that all of the prerequisites detailed in *Prerequisites* on page 16 are met.

Install Configuration Studio

To install Configuration Studio in GUI mode, perform an installation following the steps described in *Installation options* on page 28, using the following selections:

- Select the **Custom** setup type. This screen is omitted on Windows.
- Select to install the Configuration Studio component.

To install Configuration Studio in unattended mode, follow the steps described in *Unattended installation* on page 33.

The following example shows how to install the Configuration Studio component in unattended mode on Linux:

```
./APIGateway_7.6.2_Install_linux-x86-32_BN<n>.run --mode unattended
--setup_type advanced
--enable-components configurationstudio
--disable-components nodemanager,apigateway,qstart,apitester,
analytics,policystudio,apimgmt,cassandra,packagedeploytools
```

Start Configuration Studio

To start Configuration Studio after installation, perform the following steps:

1. Open a command prompt.
2. Change to your Configuration Studio installation directory (for example, `INSTALL_DIR/configurationstudio`).
3. Run `configurationstudio`.

Install API Tester

API Tester is a graphical tool on Linux that enables you to test API functionality, performance, and security. For more details on API Gateway components and concepts, see the *API Gateway Concepts Guide* .

Note API Tester is deprecated and will be removed in a future release. API Tester is no longer installed in a **Standard** or **Complete** setup, and is only installed in a **Custom** setup.

Prerequisites

Ensure that all of the prerequisites detailed in *Prerequisites* on page 16 are met.

Install API Tester

To install API Tester in GUI mode, perform an installation following the steps described in *Installation options* on page 28, using the following selections:

- Select the **Custom** setup type.
- Select to install the API Tester component.

To install API Tester in unattended mode, follow the steps described in *Unattended installation* on page 33.

The following example shows how to install the API Tester component in unattended mode:

```
./APIGateway_7.6.2_Install_linux-x86-32_BN<n>.run --mode unattended
```

```
--setup_type advanced
--enable-components apitester
--disable-components nodemanager,apigateway,qstart,policystudio,
analytics,configurationstudio,apimgmt,cassandra,packagedeploytools
```

Start API Tester

Note Before starting API Tester, ensure that the Admin Node Manager and the API Gateway instance are running. For more details, see *Start API Gateway* on page 40.

To start API Tester after installation, perform the following steps:

1. Open a command prompt.
2. Change to your API Tester installation directory (for example, `INSTALL_DIR/apitester`).
3. Run `apitester`.

For more details on API Tester, see the *API Tester User Guide* available from Axway Support at https://support.axway.com.

Install API Manager

API Manager is an additional licensed layered product running on the Axway API Gateway. For more details, see the *API Manager User Guide*.

Note Windows is supported only for a limited set of developer tools, see *Install developer tools on Windows* on page 60. API Gateway and API Manager do not support Windows.

Prerequisites

Ensure that all of the prerequisites detailed in *Prerequisites* on page 16 are met.

Axway license file

You must have a valid Axway license file to install API Manager. To obtain an evaluation trial license or a full license, contact your Axway Account Manager.

Note Your API Gateway installation must also be licensed. If you do not have a license for API Gateway, you cannot install API Manager.

Domains with multiple nodes

In an API Gateway domain environment with multiple machine nodes, API Manager must be installed on API Gateway instance nodes.

Apache Cassandra

The Apache Cassandra database is required to store API Manager data. You can install Cassandra separately before installing API Manager, or when installing API Manager. For more details, see topic '*Install an Apache Cassandra database*' in the *Install API Gateway* on page 27 section.

Install API Manager

To install API Manager in GUI mode, perform an installation following the steps described in *Installation options* on page 28, using the following selections:

- Select the **Custom** setup type.
- Select to install the following components:
    - API Manager
    - API Gateway Server
    - Admin Node Manager
    - Cassandra (if not already installed separately before API Manager)

For more details, see the following:

- *Install the API Gateway server* on page 39
- *Install the Admin Node Manager* on page 44
- *Install an Apache Cassandra database* on page 36

Unattended mode

To install API Manager in unattended mode, follow the steps described in *Unattended installation* on page 33.

The following example shows how to install the API Manager, API Gateway Server, Admin Node Manager, and Cassandra components in unattended mode:

```
./APIGateway_7.6.2_Install_linux-x86-32_BN<n>.run --mode unattended
--setup_type advanced
--enable-components apimgmt,apigateway,nodemanager,cassandra
--disable-components qstart,policystudio,configurationstudio,
analytics,apitester,packagedeploytools
--licenseFilePath mylicense.lic
--apimgmtLicenseFilePath mymgmtlicense.lic
```

Configure API Manager

If you selected to install the QuickStart tutorial, API Manager is configured by default. If you did not install the QuickStart tutorial, you must configure API Manager. For more details, see the *API Manager User Guide*.

Start API Manager

Note Before starting API Manager, ensure that Apache Cassandra, the Admin Node Manager and API Gateway instance are running. For more details, see *Start API Gateway* on page 40.

When API Manager is configured, you can use the following URL to log into the API Manager web console:

```
https://HOST:8075
```

The default URL is:

```
https://localhost:8075
```

Enter your API administrator user credentials. This is the API administrator user name and password you entered during installation.

For more information on using API Manager, see the *API Manager User Guide*.

Install the Package and Deploy tools

You can use the API Gateway Package and Deploy tools to automate processes in your API Gateway system for continuous integration. For example, this includes generating API Gateway configuration packages from API team development projects, and building and deploying configurations to API Gateway group instances.

You can install the Package and Deploy tools component on both Linux and Windows without an API Gateway license.

Note Windows is supported only for a limited set of developer tools, see *Install developer tools on Windows* on page 60. API Gateway and API Manager do not support Windows.

For more details on API Gateway configuration packages, see the *API Gateway DevOps Deployment Guide*.

Prerequisites

Ensure that all of the prerequisites detailed in *Prerequisites* on page 16 are met.

Install the Package and Deploy tools

To install the Package and Deploy tools in GUI mode, perform an installation following the steps described in *Installation options* on page 28, using the following selections:

- Select the **Custom** setup type. This screen is omitted on Windows.
- Select to install the **Package and Deploy Tools** component.

To install the Package and Deploy tools component in unattended mode, follow the steps described in *Unattended installation* on page 33.

For example, the following command shows how to install the API Gateway Package and Deploy tools only in unattended mode on Linux:

```
./APIGateway_7.6.2_Install_linux-x86-32_BN<n>.run --mode unattended
--setup_type advanced
--enable-components packagedeploytools
--disable-components apigateway,qstart,policystudio,
analytics,configurationstudio,apitester,apimgmt,cassandra
```

For details on using the Package and Deploy tools to automate processes for continuous integration, see "Upgrade an API Gateway project" in the *API Gateway DevOps Deployment Guide*.

Install API Gateway Analytics

API Gateway Analytics is a server runtime and web-based console for analyzing and reporting on API use over extended periods of time. For more details, see the *API Gateway Analytics User Guide*.

Prerequisites

Ensure that all of the prerequisites detailed in *Prerequisites* on page 16 are met.

Axway license file

You must have a valid Axway license file to install API Gateway Analytics. To obtain an evaluation trial license or a full license, contact your Axway Account Manager.

Install API Gateway Analytics

To install API Gateway Analytics in GUI mode, perform an installation following the steps described in *Installation options* on page 28, using the following selections:

- Select the **Custom** setup type.
- Select to install the API Gateway Analytics component.

To install API Gateway Analytics in unattended mode, follow the steps described in *Unattended installation* on page 33.

The following example shows how to install the API Gateway Analytics component in unattended mode:

```
./APIGateway_7.6.2_Install_linux-x86-32_BN<n>.run --mode unattended
--setup_type advanced
--enable-components analytics
--disable-components nodemanager,apigateway,qstart,policystudio,
apitester,configurationstudio,apimgmt,cassandra,packagedeploytools
--analyticsLicenseFilePath myanalyticslicense.lic
```

Configure your metrics database for API Gateway Analytics

Note Before starting API Gateway Analytics, you must perform the following steps:

1. Create a database instance to store metrics for API Gateway Analytics. Alternatively, if you already have an existing database, skip to the next step.
2. Update your API Gateway Analytics configuration with the database details using the `configureserver` script.
3. Configure the database tables using the `dbsetup` script.
4. Enable writing of metrics from your API Gateway instance to the database using the `managedomain` tool.

For more details on how to perform these steps, see the *API Gateway Analytics User Guide*.

Start API Gateway Analytics

When you have configured your metrics database and API Gateway Analytics, you can start up API Gateway Analytics. For more details, see the *API Gateway Analytics User Guide*.

Start as a service

You can also run API Gateway Analytics as a service. For more information, see *Set up services* on page 64.

Further information

For details on how to set up scheduled reports, view monitoring data in API Gateway Analytics, or purge the metrics database, see the *API Gateway Analytics User Guide*.

For details on using Policy Studio to configure policies, see the *API Gateway Policy Developer Guide*.

Install and configure a metrics database 6

API Gateway stores and maintains monitoring and transaction data in a JDBC-compliant database, which can be read by API Gateway Analytics, API Manager, and third-party monitoring tools.

This topic describes how to create and configure a database for monitoring in API Manager and third-party tools.

For details on configuring a database for API Gateway Analytics, see the *API Gateway Analytics User Guide*.

Prerequisites

The prerequisites for setting up the database are as follows:

Install a third-party JDBC database

You must install a JDBC-compliant database to store the monitoring and transaction data. Axway provides setup scripts for the following databases:

- MySQL or MariaDB
- Microsoft SQL Server
- Oracle
- IBM DB2

For details on supported database versions, see *System requirements* on page 16. For details on how to install your chosen third-party JDBC database, see your database product documentation.

Note You must ensure that you have the correct credentials to execute the setup scripts and to access the database for operations on the tables created by the scripts.

Install API Manager

You must install API Manager to use it to view the monitoring data in the metrics database. For more details, see Install API Manager.

Note You do not need to install API Gateway Analytics to view monitoring data in API Manager only.

Add third-party JDBC driver files

You must add the JDBC driver files for your chosen third-party database to your API Gateway and Policy Studio installations as appropriate.

Add JDBC drivers to API Gateway

To add the third-party JDBC driver files for your database to API Gateway, perform the following steps:

1. Add the binary files for your database driver as follows:
    - Add `.jar` files to `INSTALL_DIR/apigateway/ext/lib`
    - Add `.so` files to the `INSTALL_DIR/apigateway/platform/lib`
2. Restart API Gateway.

Add JDBC drivers to Policy Studio

To add third-party binaries to Policy Studio, perform the following steps:

1. Select **Window > Preferences > Runtime Dependencies** from the Policy Studio main menu.
2. Click **Add** to select a JAR file to add to the list of dependencies.
3. Click **Apply** when finished. A copy of the JAR file is added to the `plugins` directory in your Policy Studio installation.
4. Click **OK**.
5. Restart Policy Studio using the `policystudio -clean` command.

Create the third-party database

API Manager monitoring reads message metrics from a third-party JDBC database and display this information in a visual format to administrators. This is the same database in which API Gateway stores its message metrics and audit trail data. You must first create this database using the third-party database of your choice:

- MySQL or MariaDB
- Microsoft SQL Server
- Oracle
- IBM DB2

For details on how to do this, see the product documentation for your chosen third-party database. The following example shows creating a MySQL or MariaDB database:

```
mysql> CREATE DATABASE reports;
Query OK, 1 row affected (0.00 sec)
```

In this example, the metrics database is named `reports`, but you can use any appropriate name.

Set transaction isolation to READ COMMITTED

For all supported databases, to ensure atomicity and consistency, you must ensure that the transaction isolation level is set to `READ COMMITTED`. This setting is recommended whether you are installing for the first time or upgrading.

Note Read-committed transaction isolation mode is the default mode for Oracle, Microsoft SQL Server and IBM DB2, but not for MySQL or MariaDB. If you are using MySQL or MariaDB, you must change to read-committed transaction isolation mode after installation and before you start the server for the first time.

For more details, see the product documentation for your chosen third-party database.

Configure the database connection

You must ensure that the API Gateway external connection to the database has been configured in Policy Studio. To configure a connection, select **Environment Configuration** > **External Connections** > **Database Connections** > **Add a Database Connection**. For more details, see the *API Gateway Policy Developer Filter Reference*.

Set up the database tables

For API Manager monitoring, run the `dbsetup` command from the following API Gateway directory:

```
INSTALL_DIR/apigateway/posix/bin
```

The following example command shows setting up new database tables:

```
dbsetup
New database
Schema successfully upgraded to:002-leaf
```

Note When you specify command-line arguments to `dbsetup`, the script does not run interactively. You should run `dbsetup` without any options to create the database tables.

Specify options to dbsetup

You can specify the following options to the `dbsetup` command:

Option	Description
`-h, --help`	Displays help message and exits.
`-p PASSPHRASE, --passphrase=PASSPHRASE`	Specifies the configuration passphrase (blank for zero length).
`--dbname=DBNAME`	Specifies the database name (mutually exclusive with `--dburl`,`--dbuser`, and `--dbpass`).
`--dburl=DBURL`	Specifies the database URL.
`--dbuser=DBUSER`	Specifies the database user.
`--dbpass=DBPASS`	Specifies the database password. You must enclose passwords that contain special characters in single quotation marks. For example: `./dbsetup --dburl=mysql://127.0.0.1:3306/reports --dbuser=root --dbpass='AcmeCorp!23'`
`--reinstall`	Forces a reinstall of the database, dropping all data.
`--stop=STOP`	Stops the database upgrade after the named upgrade.

dbsetup examples

The following are some examples of using `dbsetup` command options.

Connect to a named database

You can use the `--dbname` option to connect to a named database connection configured under the **External Connections** node in the Policy Studio tree. For example:

```
dbsetup --dbname=Oracle
Current schema version:001-initial
Latest schema version:002-leaf
Schema successfully upgraded to:002-leaf
```

Connect to a database URL

You can use the `--dburl` option to manually connect to a database instance directly using a URL. For example:

```
dbsetup --dburl=jdbc:mysql://localhost/reports --dbuser=root --dbpass=admin
Current schema version:001-initial
Latest schema version:002-leaf
Schema successfully upgraded to:002-leaf
```

Install a database

You can also use the `--dburl` option to set up a newly created database instance where none already exists. For example:

```
dbsetup --dburl=jdbc:mysql://localhost/reports --dbuser=root --dbpass=admin
New database
Schema successfully upgraded to:002-leaf
```

Reinstall a database

You can use the `--reinstall` option to wipe and reinstall a database. For example:

```
dbsetup --dburl=jdbc:mysql://localhost/reports --dbuser=root --dbpass=admin --
reinstall
Re-installing database...
Schema successfully upgraded to:002-leaf
```

SQL database schema scripts

As an alternative to using the `dbsetup` command, API Gateway also provides separate SQL schema scripts to set up the database tables for each of the supported databases. However, these scripts set up new tables only, and do not perform any upgrades of existing tables. These scripts are provided in the `INSTALL_DIR/apigateway/system/conf/sql` directory in the following sub-directories:

- `/mysql`
- `/mssql`
- `/oracle`
- `/db2`

Note The scripts in the `/mysql` folder apply to both MySQL and MariaDB.

You can run the SQL commands in the `analytics.sql` file in the appropriate directory for your database. The following example shows creating the tables for a MySQL database:

```
mysql> \. INSTALL_DIR/apigateway/system/conf/sql/mysql/analytics.sql
Query OK, 0 rows affected, 1 warning (0.00 sec)
Query OK, 0 rows affected, 1 warning (0.00 sec)
...
```

Further information

For details on how to view monitoring metrics in API Manager, see the *API Manager User Guide*.

For details on how to view monitoring metrics in API Gateway Analytics, see the *API Gateway Analytics User Guide*.

Install developer tools on Windows 7

Windows is supported only for a limited set of developer tools that can be used to manage an an existing deployment running on Linux or in Docker containers. API Gateway and API Manager do not support Windows.

The following tools can be installed on Windows:

- Policy Studio
- Configuration Studio
- Package and Deployment Tools

You can install the developer tools using the same installer flow as on Linux. To run the installer in the default GUI mode, locate and run the following setup file:

```
APIGateway_7.6.2_Client_Tools_Install_win-x86-32_BN<n>.exe
```

Because some options do not apply on Windows installation, screens relating to these options are omitted. The components available for Windows can be installed without a license.

You can choose to install the available developers tools either together or separately. Unattended installation is also available. For more details on each of the tools, see the following:

- *Install Configuration Studio* on page 46
- *Install Policy Studio* on page 45
- *Install the Package and Deploy tools* on page 50

Post-installation 8

This topic describes various tasks that you might perform after installing API Gateway. This includes how to check if an installation has been successful, any initial configuration needed before you can start API Gateway, what you should do to secure API Gateway, and so on.

This topic includes the following:

Verify the installation

To verify your installation, follow these guidelines:

- Check the installation results
- Start API Gateway components
- Log in to the API Gateway tools

Check the installation log

You can examine the installation log in the root directory of the installation (for example, `Axway-installLog.log`).

Start API Gateway components

- To start the API Gateway server and Admin Node Manager, see *Start API Gateway* on page 40.
- To start the API Gateway Analytics server, see the *API Gateway Analytics User Guide*.

Log in to the API Gateway tools

- To start the Policy Studio desktop tool, see *Start Policy Studio* on page 45.
- To log in to the API Gateway Manager web-based administration tool, see *Start API Gateway* on page 40.
- To start the Configuration Studio desktop tool, see *Start Configuration Studio* on page 47.
- To start the API Tester desktop tool, see *Start API Tester* on page 48.
- To log in to the API Manager web-based tool, see *Start API Manager* on page 50.
- To log in to the API Gateway Analytics web-based tool, see the *API Gateway Analytics User Guide*.

Initial configuration

Depending on the installation options you selected, the following tasks might need to be completed before you can start API Gateway.

Create a new domain

If you did not install the QuickStart tutorial, you must use the `managedomain` script to create a new managed domain that includes an API Gateway instance. You can run the script from the following directory

```
INSTALL_DIR/apigateway/posix/bin
```

For more details on running `managedomain`, see "Configure an API Gateway domain" in the *API Gateway Administrator Guide*.

Run API Gateway on privileged ports

API Gateway is run as a non-root user to prevent any potential security issues with running as the `root` user. To enable API Gateway to listen on privileged ports when running as non-root, you must perform the steps in "Run API Gateway on privileged ports" in the *API Gateway Administrator Guide*. If you do not perform these steps, the following error is reported during API Gateway startup:

```
ERROR   ...  failed to listen on address 0.0.0.0/80: Permission denied. can't bind
socket to address
```

Set up a database for API Gateway Analytics

If you installed API Gateway Analytics, you must set up a JDBC-compliant database, before you can start API Gateway Analytics:

- First, you must install and configure a database to store the monitoring and transaction data read by API Gateway Analytics.
- Next, you must configure API Gateway Analytics to use this database instead of the default MySQL database stored on the local machine.

For more details, see the *API Gateway Analytics User Guide*.

Alternatively, if you installed API Manager, see *Install and configure a metrics database* on page 54.

Secure API Gateway

Perform the following tasks after installation to secure your API Gateway system and protect the API Gateway environment from internal or external threats.

Change default passwords

If you did not set an administrator user name and password during installation, you should change the default administrator user name and password now. For details, see "Manage administrator users" in the *API Gateway Administrator Guide*.

Change default certificates

The default certificates used to secure API Gateway components are self-signed. You can replace these self-signed certificates with certificates issued by a Certificate Authority (CA). For details, see "Manage certificates and keys" in the *API Gateway Administrator Guide*.

Encrypt API Gateway configuration

By default, API Gateway configuration is unencrypted. You can specify a passphrase to encrypt API Gateway instance configuration as detailed in "Configure an API Gateway encryption passphrase" in the *API Gateway Administrator Guide*.

Change default session timeout for API Gateway Manager

The default idle session timeout for the API Gateway Manager web UI is 12 hours. It is recommended that you change this timeout to 120 minutes or less:

1. Open the file `INSTALL_DIR/apigateway/conf/envSettings.props`.
2. Edit the property `env.WEBMANAGER.SESSION.TIMEOUT`. The property value is in milliseconds. The default value is 43200000 (12 hours).
3. Restart API Gateway for the updates to be applied.

Where to go next

For additional procedures you can perform to secure your API Gateway, see "Manage API Gateway security" in the *API Gateway Administrator Guide* .

For more information on the security features of API Gateway and best practices for strengthening the security of API Gateway, see the *API Management Security Guide*.

Set up services

This section explains how to run various components as services.

API Gateway

You can run Node Managers and API Gateway instances as services using the `managedomain` script. To register a Node Manager or an API Gateway instance as a service on Linux, you must run the `managedomain` command as `root`. For example:

- Node Manager: Enter `managedomain --menu`, and choose option 2, `Edit a host`.
- API Gateway instance: Enter `managedomain --menu`, and choose option 10, `Add script or service for existing local API Gateway`.

Alternatively, you can run `managedomain` in command mode with the `--add_service` option to create a service for a Node Manager or API Gateway instance.

For more details on `managedomain`, see the *API Gateway Administrator Guide*.

API Gateway Analytics

You can also run the API Gateway Analytics server as a service by creating a script. A sample script and *ReadMe* is provided in the following directory:

```
INSTALL_DIR/analytics/posix/samples/etc/init.d/
```

Apache Cassandra

For details on running Apache Cassandra as a service, see *Install an Apache Cassandra database* on page 36.

Set up clustering

To set up API Gateway for high availability, you need to configure an external Apache Cassandra database for clustering. For more information, see the following topics:

- "Configure a Cassandra HA cluster" in the *API Gateway Apache Cassandra Administrator Guide*
- *Configure API Management in multiple datacenters* on page 66

Next steps

Consult the *API Gateway Administrator Guide* for more information on administering, managing, and troubleshooting an API Gateway system. This guide contains many topics that you will find useful after installing API Gateway. For example:

- Manage an API Gateway domain
- Configure API Gateway for high availability
- Backup and disaster recovery
- Manage user access

Configure API Management in multiple datacenters 9

This topic describes the recommended multi-datacenter configuration that applies to the various types of API Management data in storage. For each data type, it describes how data is replicated across the datacenter, the recommended configuration, and expected behavior in case of failover.

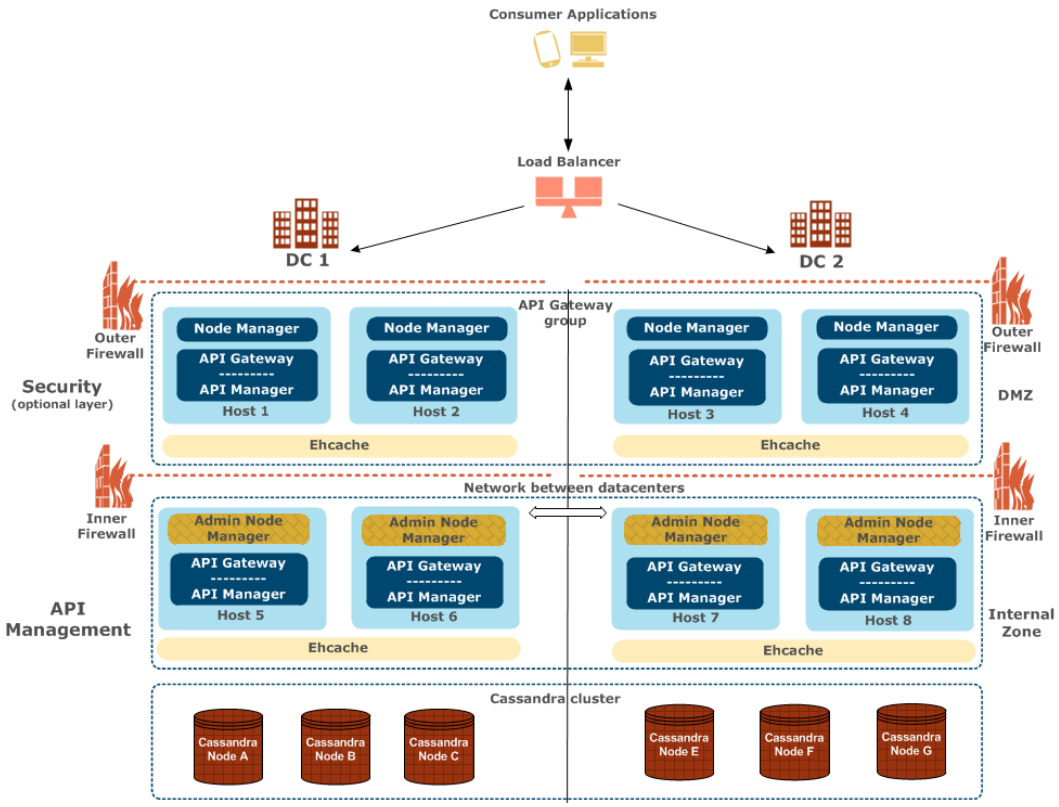
This topic includes the following:

Multi-datacenter deployment

This topic describes the infrastructure required for API Management multi-datacenter deployment. It describes the various types of API Management data. For example, this includes API catalog, client registry, OAuth tokens, quota, Key Property Store (KPS), and so on. It also describes where the data is stored. For example, this includes files on disk, Apache Cassandra database, Ehcache, or Relational Database Management System (RDBMS). For details on supported database versions, see *System requirements* on page 16.

Multi-datacenter deployment architecture

The following diagram shows the minimum infrastructure required for API Management multi-datacenter deployment.



This deployment architecture is described as follows:

- Each datacenter includes the same components deployed in active/active mode. Each datacenter can handle all of the traffic load and can scale in the same way. The API Gateway instances and Cassandra nodes in each datacenter are all highly available.
- API Gateway group configuration is shared across both datacenters. This means that all API Gateway instances in a group are managed as a single unit, and run the same configuration to virtualize the same APIs and execute the same policies.
- You must have at least two API Gateway instances per datacenter, at least one of which is an Admin Node Manager. However, you can configure mutliple Admin Node Managers per datacenter for high availability. For more details, see the *API Gateway Administrator Guide*.
- The API Gateway group in the internal zone is responsible for API management. The Admin Node Manager is the central administration server responsible for all management operations. For example, this enables API Gateway administrators to perform monitoring in the API Gateway Manager web console.
- The API Gateway group in the internal zone also hosts the API Manager web console used by API administrators. For more details, see the *API Manager User Guide*.
- The API Gateway group in the outer DMZ is responsible for securing traffic. The Node Manager manages local API Gateways instances on that host only.
- The Cassandra database cluster is required to store data for API Manager or API Gateway client registry (API key or OAuth). You can use Cassandra as an option to store custom KPS data and OAuth tokens. You can also use an RDBMS to store custom KPS, OAuth tokens, or metrics for API Manager or API Gateway Analytics.

- Caching is replicated between API Gateway instances using the Ehcache distributed caching system. For more details, see "Global caches" in the *API Gateway Policy Developer Guide*.

API Management data storage

This section describes what API Management data can be persisted and where.

API Gateway data

Data type	Storage location
API Gateway configuration	Files on disk: • API Gateway instance: `INSTALL_DIR/apigateway/groups/group-n/instance-n/conf/fed` • Node Manager/Admin Node Manager: `INSTALL_DIR/apigateway/conf/fed` Alternatively, you can use a deployment archive (`.fed` file). For more details, see the *API Gateway DevOps Deployment Guide*.
API Gateway logs	Files on disk: • API Gateway instance: `INSTALL_DIR/apigateway/groups/group-n/instance-n/logs` • Node Manager/Admin Node Manager: `INSTALL_DIR/apigateway/logs`
API Gateway traffic monitoring	Files on disk: • API Gateway instance: `INSTALL_DIR/apigateway/groups/group-n/instance-n/conf/opsdb.d` • Node Manager/Admin Node Manager: `INSTALL_DIR/apigateway/conf/opsdb.d`
API Gateway KPS custom tables	Cassandra or RDBMS
API Gateway OAuth token store	Ehcache, Cassandra, or RDBMS.

Data type	Storage location
API Gateway throttling counters	Ehcache
API Gateway custom cache	Ehcache

API Manager data

Data type	Storage location
API Manager catalog, client registry, web-based settings	Cassandra
API Manager quota counters	In memory, Cassandra, or RDBMS
API Manager metrics	RDBMS

Further details

For more details on concepts such as shared group configuration, Node Manager, and Admin Node Manager, see the *API Gateway Concepts Guide*.

For details on how to configure API Management in multiple datacenters, see

- *Multi-datacenter configuration* on page 69
- *Multi-datacenter failover scenarios* on page 87

Multi-datacenter configuration

This topic describes the recommended configuration for each API Management data type. It explains the following for each data type:

- If the data can be replicated across both datacenters (for example, by deploying in both datacenters or automatic replication)
- How to configure the replication (for example, by configuring Apache Cassandra, Ehcache, or RDBMS)
- How to install and configure API Gateway and API Manager in multiple datacenters, and how to optimize performance

For details on recommended architecture, see *Multi-datacenter deployment* on page 66.

API Management configuration

The recommended configuration for each data type is as follows:

API Gateway data

Data type	Replication between datacenters
API Gateway configuration	You must deploy the API Gateway group configuration in both datacenters. For details on automating deployment processes, see the *API Gateway DevOps Deployment Guide*.
API Gateway logs	Does not apply (local file-based data only).
API Gateway traffic monitoring	Does not apply (local file-based data only).
API Gateway KPS custom tables	Automatic. It is best to use Cassandra with replication between datacenters. See *Configure Cassandra for multiple datacenters* on page 71, or your RDBMS documentation.
API Gateway OAuth token store	It is best to use Ehcache, and to generate and use OAuth tokens in the same datacenter only. You should configure sticky sessions in your load balancer. See *Configure Ehcache in multiple datacenters* on page 84. See *Configure Ehcache in multiple datacenters* on page 84.
API Gateway throttling counters	It is best to configure at least one distributed cache per datacenter and to avoid replication between datacenters. You should configure sticky sessions in your load balancer. See *Configure Ehcache in multiple datacenters* on page 84.
API Gateway custom cache	It is best to configure at least one distributed cache per datacenter and to avoid replication between datacenters. You should configure sticky sessions in your load balancer. See *Configure Ehcache in multiple datacenters* on page 84.

API Manager data

Data type	Replication between datacenters
API Manager catalog, client registry, web-based settings	Automatic. It is best to use Cassandra with replication between datacenters. See *Configure Cassandra for multiple datacenters* on page 71.

Data type	Replication between datacenters
API Manager quota counters	In memory only, or automatic when using external storage (Cassandra or RDBMS). See *Configure API Manager quota in multiple datacenters* on page 87. See also *Configure Cassandra for multiple datacenters* on page 71, or your RDBMS documentation.
API Manager metrics	Automatic. See your RDBMS documentation.

Configure Cassandra for multiple datacenters

Cassandra is required to store data for API Manager data, and also recommended for API Gateway custom KPS tables. For details on the recommended Cassandra architecture, see *Multi-datacenter deployment architecture* on page 66.

Note You must install and configure Cassandra on each node in both datacenters before installing and configuring API Gateway and API Manager.

Prerequisites

The following prerequisites apply to Cassandra in a multi-datacenter production environment:

- Ensure that Cassandra version 2.2.12 is installed. For details, see *Install an Apache Cassandra database* on page 36. For details on upgrading Cassandra, see "Upgrade Apache Cassandra" in the *API Gateway Upgrade Guide*.
- You must have at least three Cassandra nodes per datacenter. Cassandra must be installed on each node in the cluster, but should not be started until the Cassandra cluster is fully configured. For more details, see *Install an Apache Cassandra database* on page 36.
- Configure `JAVA_HOME` to an Oracle JRE 1.8 installation (OpenJDK is not supported).
- Each Cassandra node must have Python 2.7.x installed.
- Time must be synchronized on all servers.
- Determine a naming convention for each datacenter and rack, for example:
  - `DC1`, `DC2`
  - `RACK1`, `RACK2`, `RACK3`

    Caution The rack names must be exactly the same in each datacenter. You should choose the names carefully because renaming a datacenter is not possible.
- Determine the seed nodes. You must have at least two Cassandra seed nodes per datacenter.
- Choose a unique name for the Cassandra cluster.

- To avoid firewall issues, you must open the following ports to allow bi-directional communication among the nodes:
  - `7001`: Cassandra TLS/SSL inter-node cluster communication
  - `7199`: Cassandra JMX monitoring port (only enabled on `localhost` for security reasons, you must use SSH to connect the machine).
  - `9042`: CQL native client port (only required for an API Gateway or API Manager client connection to Cassandra)

Configure your cassandra.yaml file

On each node in the cluster, you must configure the following properties in the `CASSANDRA_HOME/cassandra/conf/cassandra.yaml` file:

Configure multi-datacenter settings

Configure the following settings in `cassandra.yaml`:

- `cluster_name`: Your chosen cluster name, common across both datacenters (for example, `cassandra-cluster1`)
- `num_tokens`: `256`
- `endpoint_snitch`: `GossipingPropertyFileSnitch`
- `-seeds`: Internal IP address for all seed nodes. Each datacenter should have its own seed node IP addresses first, followed by the other datacenters seed nodes. This is critical for replication.

  For example:
  - DC1 seed nodes: `192.168.10.1,192.168.10.2`
  - DC2 seed nodes: `192.168.20.1,192.168.20.2`
  - DC1 configuration:
    `-seeds: 192.168.10.1,192.168.10.2,192.168.20.1, 192.168.20.2`
  - DC2 configuration:
    `- seeds: 192.168.20.1,192.168.20.2,192.168.10.1, 192.168.10.2`
- `listen_address`: Specify the *private* IP address of the current node that you are configuring.
- `broadcast_address`: Specify the *public* IP address of the node. For example, this is important when using a VPN with Network Address Translation (NAT) of IP addresses. Communication between datacenters can only take place using an external IP address.
- `start_native_transport`: `true`
- `native_transport_port`: `9042`

Note You must remove `cassandra-topology.properties` from `cassandra/conf/`. This is not needed when `GossipingPropertyFileSnitch` is set and could cause conflicts.

Configure authentication settings

It is best to enable authentication. Set the following properties:

- `authenticator: org.apache.cassandra.auth.PasswordAuthenticator`
- `authorizer: org.apache.cassandra.auth.CassandraAuthorizer`

Note When these properties are set, you must change the default Cassandra user.

Configure TLS/SSL traffic encryption

It is best to enable inter-node and client-to-node TLS/SSL encryption. API Management supports TLS v1.2.

Node-to-node encryption

To enable node-to-node TLS/SSL traffic encryption, set the following properties:

`server_encryption_options:`

- `internode_encryption: all`
- `keystore: my-server-keystore.jks`
- `keystore_password: MY_KEYSTORE_PASSWORD`
- `truststore: my-server-truststore.jks`
- `truststore_password: MY_TRUSTSTORE_PASSWORD`
- `require_client_auth: true`

Client-to-node encryption

To enable client-to-node SSL traffic encryption, set the following properties:

`client_encryption_options:`

- `enabled: true`
- `optional: false`
- `keystore: my-client-keystore.jks`
- `keystore_password: MY_KEYSTORE_PASSWORD`
- `truststore: my-client-truststore.jks`
- `truststore_password: MY_TRUSTSTORE_PASSWORD`
- `require_client_auth: true`

Configure your cassandra-rackdc.properties file

Configure the `cassandra/conf/cassandra-rackdc.properties` file with your chosen datacenter and rack names for each node, and set `prefer_local=true`.

Caution The rack names must be exactly the same in each datacenter. You should choose the names carefully because renaming a datacenter is not possible.

For example:

Node	Setting in cassandra-rackdc.properties

DC1, node1	`dc=DC1`
	`rack=RACK1`
	`prefer_local=true`

DC1, node2	`dc=DC1`
	`rack=RACK2`
	`prefer_local=true`

DC1, node3	`dc=DC1`
	`rack=RACK3`
	`prefer_local=true`

DC2, node1	`dc=DC2`
	`rack=RACK1`
	`prefer_local=true`

DC2, node2	`dc=DC2`
	`rack=RACK2`
	`prefer_local=true`

DC2, node3	`dc=DC2`
	`rack=RACK3`
	`prefer_local=true`

Start your Cassandra nodes

When you have configured `cassandra.yaml` and `cassandra-rackdc.properties` on all nodes, you can start the seed nodes one at a time, followed by the remaining nodes in each datacenter.

For details on starting Cassandra, see *Manage Apache Cassandra* in the *API Gateway Apache Cassandra Administrator Guide*.

Configure cqlsh for TSL/SSL encryption

If the Cassandra cluster has been configured to use client-to-node TSL/SSL encryption, you must configure all clients connecting to the cluster (including `cqlsh`) to use TSL/SSL. `cqlsh` is a Python-based command line client for executing Cassandra Query Language (CQL) commands.

To configure `cqlsh` for TSL/SSL, you must provide the following certificates in a `cqlshrc` file:

- `certfile.pem`: Contains the CA or server certificate of the Cassandra node. This is specified in `my-client-keystore.jks`.
- `usercert.pem`: Contains the client certificate for `cqlsh`. This needs to be added to the Cassandra truststore: `my-client-truststore.jks`
- `userkey.pem`: Contains the private key of client certificate for `cqlsh`. This cannot contain a passphrase.

A sample is provided in `cassandra/conf/cqlshrc.sample`:

When TSL/SSL has been configured, run the following command:

```
./cqlsh -u cassandra -p MY_PASSWORD --ssl --cqlshrc=MY_CQLSHRC_FILE
192.168.10.1
```

Configure the default system_auth keyspace

When all nodes are online, if authentication is enabled, you must set the replication strategy and replication factor for the `system_auth` keyspace to ensure that credentials are shared across the cluster. You can do this using the `cqlsh` tool from the `cassandra/bin` directory.

Note You must set the replication strategy to `NetworkTopologyStrategy`, and ensure the replication factor is set to the number of nodes per datacenter (for example, `3`).

Naming is case sensitive and the keyspace definition must use the snitch-configured datacenter names used in `cassandra-rackdc.properties`.

For example, on Cassandra node 1 in DC1, perform the following steps:

1. Navigate to the `cassandra/bin` directory
2. Log in to `cqlsh` using the IP address of the current node. For example:

```
./cqlsh -u cassandra -p cassandra 192.168.10.1
```

3. Run the following command:

```
ALTER KEYSPACE "system_auth" WITH REPLICATION = {'class' :
'NetworkTopologyStrategy', 'DC1' : n, 'DC2' : n};
```

In this example, *n* is the number of nodes per datacenter.

4. Run `nodetool repair` on each node as follows:

```
./nodetool repair system_auth
```

5. Change the default Cassandra user. For more details, see the Configuring authentication documentation.

Configure API Management in multiple datacenters

When the Cassandra cluster has been set up, you can proceed with installing API Gateway and API Manager. You must have at least two API Gateway instances per datacenter.

Configure the first API Gateway node

On the first API Gateway host in DC1, perform the following steps:

1. Install API Gateway and API Manager using the API Gateway installer. Do not select Cassandra, which has already been installed. For more details, see *Install API Manager* on page 48.
2. Register the Admin Node Manager using the `managedomain` command in `INSTALL_DIR/apigateway/posix/bin`. For details, see the *API Gateway Administrator Guide*.
3. Start the Admin Node Manager using the `nodemanager` command.
4. Add the API Gateway instance and group using the `managedomain` command.
5. Before starting the API Gateway instance, add the Cassandra host names, IP addresses, and initial keyspace settings specific for a multi-datacenter environment as environment variables in your `envSettings.props` file located in `INSTALL_DIR/apigateway/conf`. For example:

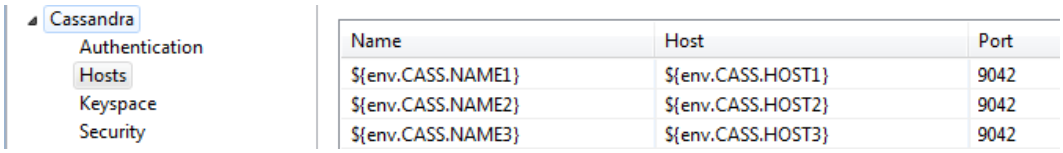
```
env.CASS.NAME1=Host 10.1
env.CASS.NAME2=Host 10.2
env.CASS.NAME3=Host 10.3
env.CASS.HOST1=192.168.10.1
env.CASS.HOST2=192.168.10.2
env.CASS.HOST3=192.168.10.3
env.CASS.REPL.FACTOR=DC1:3;DC2:3;
```

6. Start the API Gateway using the `startinstance` command in `INSTALL_DIR/apigateway/posix/bin`.
7. Configure the API Gateway to connect to the Cassandra cluster. In the Policy Studio tree, select **Server Settings > Cassandra**, and configure the following:

Keyspace:

- **Keyspace name:** Name of the API Gateway Cassandra keyspace to be created when deployed. Defaults to `x${DOMAINID}_${GROUPID}`.

- **Initial replication strategy:** Network Topology Strategy
- **Initial replication:** ${env.CASS.REPL.FACTOR}
- **Hosts**: Add the environment variable settings that you set in `envSettings.props`. For example:



- **Authentication**: Enter the Cassandra user name and password that you configured earlier (see *Configure the default system_auth keyspace* on page 75).
- **Security**: Select **Enable SSL**, and select a trusted certificate and client certificate. For example:

8. Select **File > Configure API Manager** to configure API Manager settings. For more details on configuring API Manager in Policy Studio, see the *API Manager User Guide*.
9. For all KPS collections, update the read and write consistency levels to **LOCAL_QUORUM**. For example, in the Policy Studio tree, select **Environment Configuration > Key Property Stores > API Server > Data Sources > Cassandra Storage**, and click **Edit**.

   Repeat this step for each KPS collection using Cassandra (for example, **Key Property Stores > OAuth** and **API Portal** for API Manager). This also applies to any custom KPS collections that you have created.
10. Click **Deploy** in the toolbar to deploy the updated configuration.

Note Policy Studio may need a longer transaction timeout in the Admin Nodemanager server settings than the default time (4 minutes), especially for the first deploy that creates the API Manager Cassandra tables. In this case, it is recommended to increase the timeout to at least 10 minutes. See the *API Gateway Administrator Guide* for more details. If Policy Studio shows a timeout error, the back-end would still complete and the success status can be verified in the API Manager instance trace file.

Update the Cassandra replication settings for the new API Gateway keyspace

Note This set up is only needed if the initial deployment did not set the multi-datacenter replication values.

When the new API Gateway keyspace has been deployed, you must update its replication strategy and replication factor in the same way as the default `system_auth` keyspace. This time instead of the `system_auth` keyspace name, use the name of the newly created keyspace.

Perform the following steps:

1. On Cassandra node 1 in DC1, navigate to the `CASSANDRA_HOME/cassandra/bin` directory.
2. Log in to `cqlsh` using the IP address of the current node. For example:

```
./cqlsh 192.168.10.1
```

3. Run the following command:

```
ALTER KEYSPACE "KEYSPACE_NAME" WITH REPLICATION = {'class' :
'NetworkTopologyStrategy', 'DC1' : 3, 'DC2' : 3};
```

4. On each node, run `nodetool repair`.

Tip In a production environment, you should schedule weekly node repairs as a best practice. For more details, see "Perform essential Cassandra operations" in the *API Gateway Apache Cassandra Administrator Guide*.

Configure the remaining API Gateway nodes

When the API Gateway keyspace has been deployed and its replication updated, you can register the remaining hosts and add API Gateway instances using the `managedomain` command. Alternatively, you can do this using the API Gateway Manager web console.

Note You should always add one API Gateway instance at a time to the group.

For more details on registering hosts and adding API Gateway instances, see the *API Gateway Administrator Guide*.

Configure the API Gateway environment variables in DC1

For each additional API Gateway instance in DC1, add the following to `envSettings.props` before starting the instance:

```
env.CASS.NAME1=Host 10.1
env.CASS.NAME2=Host 10.2
env.CASS.NAME3=Host 10.3
env.CASS.HOST1=192.168.10.1
env.CASS.HOST2=192.168.10.2
env.CASS.HOST3=192.168.10.3
env.CASS.REPL.FACTOR=DC1:3;DC2:3;

# API Manager Port
env.PORT.APIPORTAL=8075
# API Manager Traffic Port
env.PORT.PORTAL.TRAFFIC=8065
```

Configure the API Gateway environment variables in DC2

For each API Gateway instance in DC2, add the following settings to `envSettings.props` before starting the instance:

```
env.CASS.NAME1=Host 20.1
```

```
env.CASS.NAME2=Host 20.2
env.CASS.NAME3=Host 20.3
env.CASS.HOST1=192.168.20.1
env.CASS.HOST2=192.168.20.2
env.CASS.HOST3=192.168.20.3
env.CASS.REPL.FACTOR=DC1:3;DC2:3;

# API Manager Port
env.PORT.APIPORTAL=8075
# API Manager Traffic Port
env.PORT.PORTAL.TRAFFIC=8065
```

Add the load balancer host to API Management whitelists

For each API Gateway host, you must add the external load balancer host to the whitelists for the Node Manager and API Manager to ensure that it will be accepted in the datacenter.

Add load balancer host to Node Manager whitelist

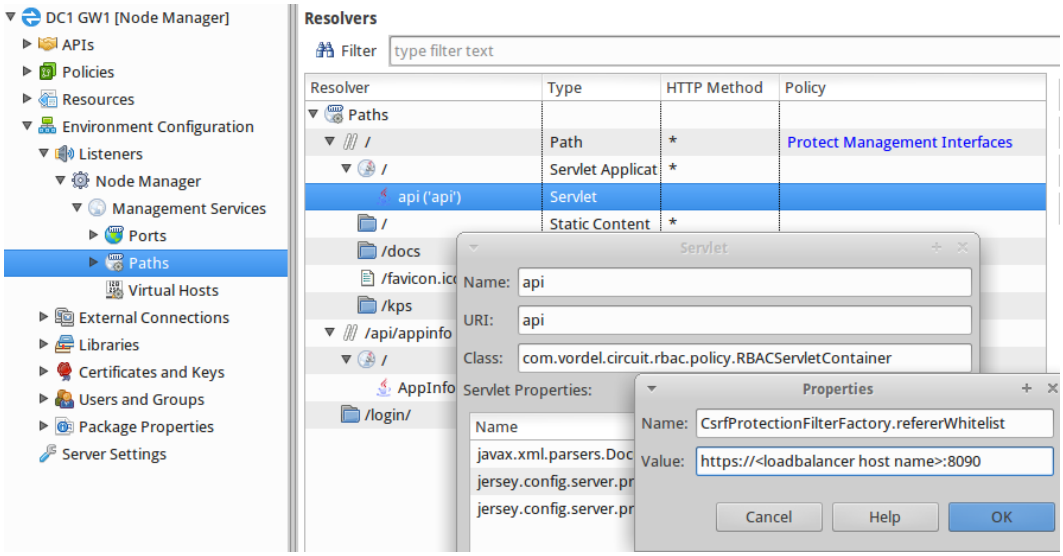
You can do this by creating an API Gateway project based on the Node Manager configuration and adding the external load balancer host to the Node Manager whitelist as a servlet property:

1. In Policy Studio, select **File > New Project**.
2. Enter a project **Name**, and click **Next**.
3. Select **From existing configuration** for the project starting point, and click **Next**.
4. Click the browse button, select the following directory, and click **Finish**:

   ```
   INSTALL_DIR/apigateway/conf/fed
   ```

5. In the Policy Studio tree, select **Environment Configuration > Listeners > Node Manager > Management Services > Paths**.
6. In the pane on the right, right-click the **api** servlet, and select **Edit**.
7. In the **Servlet** dialog, click **Add**, and enter the following in the **Properties** dialog:
   - **Name**: `CsrfProtectionFilterFactory.refererWhitelist`
   - **Value**: `https://LB_HOSTNAME:8090`
8. Right-click the **AppInfo Service** servlet, click **Edit**, and add the same whitelist setting as a property (see previous step).
9. When complete, the configuration is automatically saved to the `apiprojects` directory used by Policy Studio for your project. To get the Node Manager to read this change, you must copy the contents of your `apiprojects` subdirectory to `apigateway/conf/fed`.
10. Finally, you must restart the Node Manager for the configuration changes to be picked up.

The following shows an example of configuring the **api** servlet setting in Policy Studio:



Note You must ensure that every Node Manager has been updated to accept the load balancer host name.

You can also add multiple load balancer hosts to the whitelist. For example:

- **Name**: `CsrfProtectionFilterFactory.refererWhitelist`
- **Value**: `https://dc1-lb.example.com:8090|https://dc2-lb.example.com:8090`

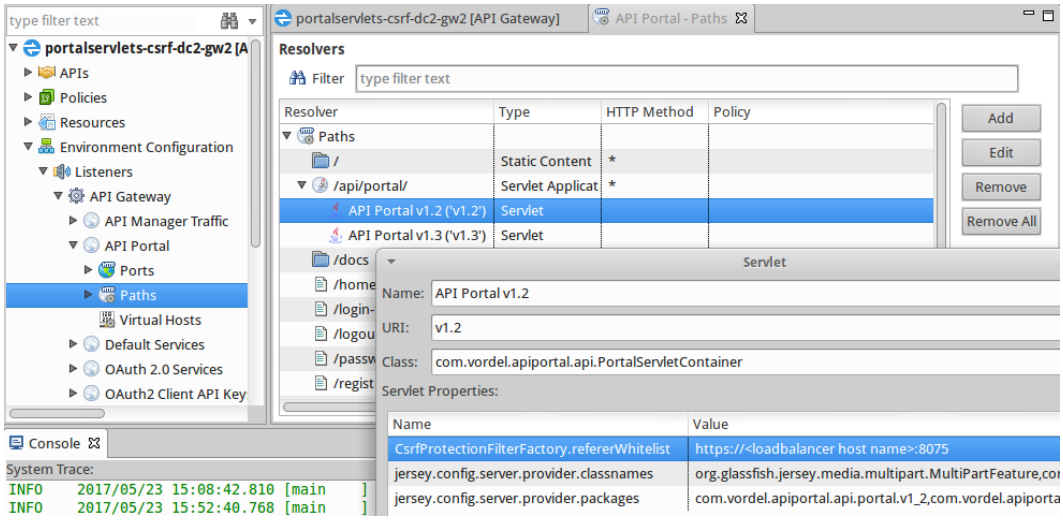
Add load balancer host to API Manager whitelist

You can do this by creating a new project from an API Gateway instance and adding the external load balancer host to the API Manager whitelist as a servlet property:

1. In Policy Studio, select **File > New Project from an API Gateway instance**.
2. Enter a project **Name**, and click **Next**.
3. In the **Connection Details** for the datacenter, enter the load balancer host name in the **Host** field and your credentials, and click **Next**.
4. Select the server instance in the datacenter that you wish to update, and click **Finish**.
5. In the Policy Studio tree, select **Environment Configuration > Listeners > API Gateway > API Portal > Paths**.
6. In the pane on the right, right-click the **API Portal v1.2 ('v1.2')** servlet, and select **Edit**.
7. In the **Servlet** dialog, click **Add** and enter the following in the **Properties** dialog:
   - **Name**: `CsrfProtectionFilterFactory.refererWhitelist`
   - **Value**: `https://<LB_HOSTNAME>:8075`
8. Right-click the **API Portal v1.3 ('v1.3')** servlet, click **Edit**, and add the same whitelist property (see previous step).

9. When complete, deploy the configuration to all API Gateways using the load balancer that you added to the whitelist.

The following shows an example of configuring the **API Portal v1.2 ('v1.2')** servlet setting in Policy Studio:



API Manager Single Sign On in multiple datacenters

For API Manager Single Sign On (SSO), when you have multiple API Managers under a specific load balancer, the `service-provider.xml` file must be the same for all API Manager instances under that load balancer. For the Keycloak identity provider, this means that the same client ID maps to the SAML profile, and that the URL should match the load balancer host name instead of the API Gateway host name.

For example, you have two datacenter sites, `apimgt.example.us` and `apimgt.example.eu`. Each site has a separate client, load balancer, and `service-provider.xml` file, and this file is shared by all API Manager instances under that load balancer.

For more details on API Manager SSO, see "Configure API Manager SSO" in the *API Manager User Guide*.

Prevent simultaneous design-time changes in multiple datacenters

If you are using API Management in two or more datacenters at design time, there is a chance that changes made to the same object simultaneously could overwrite each other and potentially put the datacenters out of sync. This is because you cannot to predict the order in which any changes are applied due to the many different variables involved.

In this case, the data would likely become inconsistent between datacenters, and it would be difficult to know that an issue had occurred. There is also no guarantee that they would correctly synchronize at any time in the future. For this reason, it is best to make any design-time changes or updates to objects in your API Management system in one datacenter only.

Optimize API Management performance in a multi-center environment

This section explains how to configure various API Gateway settings to optimize API Gateway and API Manager performance in a multi-center environment.

Increase maximum received bytes per transaction to optimize API Manager

You can configure global API Gateway settings under the **Server Settings** node in Policy Studio.

In a multi-datacenter environment with a large amount of APIs and data, you may need to increase the value of **Maximum received bytes per transaction** to optimize the performance of the API Manager web console (for example, when viewing APIs in the API catalog). The default value is `20` MB (`20971520` bytes). For example, you might need to increase this setting, depending on the number of APIs and the volume of data in your multi-datacenter environment.

To configure this setting, in the Policy Studio tree, select the **Server Settings** node, and click **General** in the right pane. To confirm updates, click **Apply changes** at the bottom right.

After changing any settings, you must deploy to the API Gateway for the changes to be enforced. Click the **Deploy** button in the toolbar, or press F6.

Increase API Manager polling time and events time for Cassandra replication

When API Manager is deployed in multiple datacenters, and changes to data stored in Cassandra are replicated across datacenters, there is a small risk that the API Gateway runtime in the datacenter where the data is replicated to might operate with outdated data. In this case, to update the data, you must restart all API Gateway instances in the affected datacenter.

To minimize this risk, you can use the `esexplorer` tool to increase the polling time in milliseconds in each API Gateway instance in the datacenter where the data is replicated to. The polling time value must be balanced with the need of having updated data in real time in the API Gateway runtime environment.

You must also change the Time To Live (TTL) of the API Manager events table to be consistent with Cassandra for failover situations. Cassandra uses a default value of three hours in its hinted handoff configuration.

To configure the API Manager polling time and events table settings, perform the following steps:

1. Change to the following directory:

   ```
   INSTALL_DIR/apigateway/posix/bin
   ```

2. Enter the `esexplorer` command.

3. Select **Store > Connect**, and browse to the following file:

```
INSTALL_DIR/apigateway/groups/<group-
name>/conf/<group-id>/configs.xml
```

4. Select **System Components > Portal Config** in the tree on the left.
5. Select the **vapiPollerPeriodMs** setting in the pane on the right. Double-click the default value of `200`, and enter a value in the range of `1000` to `30000` milliseconds.

   Note As a general rule, the higher the value of the polling time setting, the lower the risk of outdated data in the API Gateway runtime. However, it will take longer to update the data in the replicated datacenter.

6. Select the **vapiEventTTLMs** setting on the right, and double-click to enter a value of `10800000` milliseconds (3 hours). This is consistent with the default value of the `max_hint_window_in_ms` setting in the `cassandra.yaml` file.
7. When you have completed these settings, you can open the updated API Gateway project in Policy Studio and deploy the updates to all the API Gateway instances in the affected datacenter.

Increase Node Manager timeout for longer API Gateway startup

In a multi-datacenter environment, it may take longer for API Gateway to start. By default, the API Gateway active timeout is set to 4 mins. However, it may take longer for an API Gateway instance to start up in a multi-datacenter environment. For example, this may result in the Node Manager throwing a 503 error saying that the API Gateway cannot restart.

To configure the Node Manager timeout for longer startup time, perform the following steps:

1. Change to the following directory:

```
INSTALL_DIR/apigateway/posix/bin
```

2. Enter the `esexplorer` command.
3. Select **Store > Connect**, and browse to the following file:

```
INSTALL_DIR/apigateway/conf/fed/configs.xml
```

4. Select **Default System Settings** in the tree on the left.
5. Select the **activeTimeout** setting in the pane on the right. Double-click the default value of `240000` (4 minutes), and enter a higher value to better suit your multi-datacenter environment.
6. Select the **maxTransTimeout** setting on the right, and double-click to enter a higher value to suit your environment.

7. To update to all Node Managers in the group, you can copy the contents of the updated `apigateway/conf/fed` directory to the same directory on each node. Alternatively, you can run the `esexplorer` tool on each node to update the Node Manager settings.

Configure Ehcache in multiple datacenters

Caching is replicated between API Gateway instances in each datacenter using the Ehcache distributed caching system. In the distributed cache, there is no master cache controlling all caches. Instead, each cache is a peer that needs to know where all the other peers are located. The examples in this section shows distributed caches for OAuth and Throttling. The cache settings are the same in both cases.

Note You should configure at least one distributed cache per datacenter, and should not replicate Ehcache information between datacenters. You should also configure sticky sessions in your load balancer environment to optimize performance.

Configure a distributed cache in each datacenter

In a distributed cache, each API Gateway has its own local copy of the cache and registers a cache event listener, which replicates messages to the other caches. This means that events on a single cache are duplicated across all other caches.

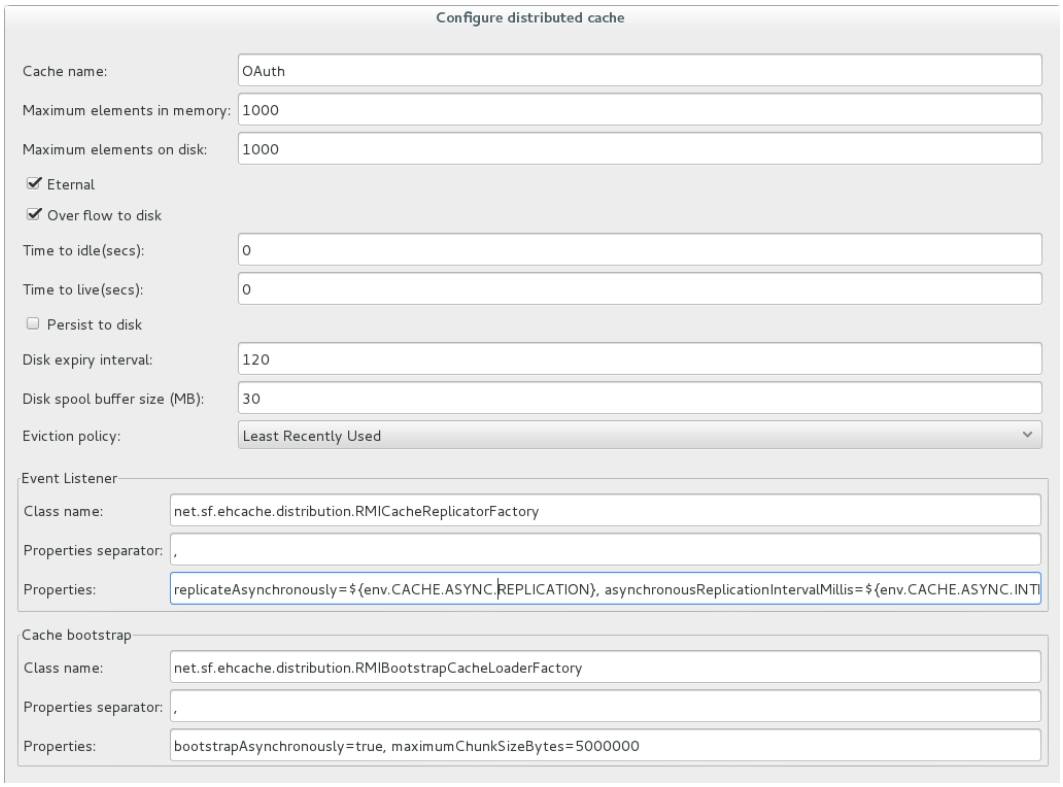
To add a distributed cache, perform the following steps:

1. Select the **Environment Configuration > Libraries > Caches** tree node, and click the **Add** button at the bottom right.
2. Select **Add Distributed Cache** from the menu, and configure the following settings on the **Configure Distributed Cache** dialog:
   - **Cache name**: Enter a name for the distributed cache (for example, `OAuth` or `Throttling`).
   - **Event Listener**: **Properties**: Enter the following setting using environment variables for the replication settings:

```
replicateAsynchronously=${env.CACHE.ASYNC.REPLICATIO
N},
asynchronousReplicationIntervalMillis=${env.CACHE.ASYN
C.INTERVAL}, replicatePuts=true,
replicateUpdates=true, replicateUpdatesViaCopy=true,
replicateRemovals=true
```

Note It is best to set `replicateAsynchronously` to true. The default `asychronousReplicationIntervaleMillis` value is `1000` ms if not specified. It is best to set this to the minimum of `10` ms.

You can leave all other settings on this dialog as default. For example, the required settings are displayed as follows in Policy Studio:



When the OAuth distributed cache has been configured, it can then be used by the OAuth Access Token Stores. For example, select **Environment Configuration > Libraries > OAuth2 Stores > Access Token Stores > OAuth Access Token Stores**, and right click to select **Edit Access Token Store**. Select **Store in a cache**, and click the browse button to select the OAuth distributed cache.

If the OAuth distributed cache is to be used by API Manager, you must also add it in **Server Settings > API Manager > OAuth Access Token Stores**.

Similarly, when a throttling distributed cache has been configured, it can then be used by the API Gateway Throttling filter. For more details, see the *API Gateway Policy Developer Filter Reference*.

Configure distributed cache settings for peer discovery in each datacenter

To configure global distributed cache settings for peer discovery, perform the following steps:

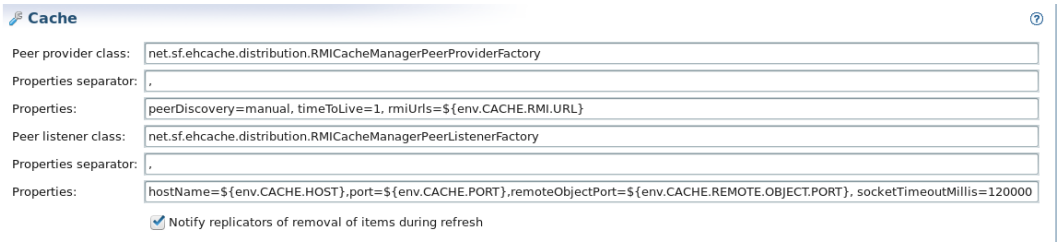
1. In Policy Studio, select the **Server Settings** node , and click **General > Cache**.
2. Configure the following settings:
    - **Peer provider class**: **Properties**: Enter the following setting using an environment variable for the cache URLs:

```
peerDiscovery=manual,timeToLive=1,rmiURLs=${env.CACHE.
RMI.URL}
```

- **Peer listener class**: **Properties**: Enter the following setting using environment variables for the hosts and ports:

```
hostName=${env.CACHE.HOST},port=${env.CACHE.PORT},remoteObject
Port=${env.CACHE.REMOTE.OBJECT.PORT}
socketTimeoutMillis=120000
```

You can leave all other settings as default. For example, the required settings are displayed as follows in Policy Studio:



Set the environment variables on each API Gateway host

You must set the environment variables used in the distributed cache in the following file on each host machine:

```
INSTALL_DIR/apigateway/conf/envSettings.props
```

For example:

```
env.CACHE.RMI.URL=//192.168.10.11:40001/OAuth|//192.168.10.11:4
0001/Throttle
env.CACHE.HOST=192.168.10.10
env.CACHE.PORT=40001
env.CACHE.REMOTE.OBJECT.PORT=40002
env.CACHE.ASYNC.REPLICATION=true
env.CACHE.ASYNC.INTERVAL=10
```

Note The `env.CACHE.RMI.URL` environment variable should only include URLs for host machines in the same datacenter.

For more details, see "Global caches" in the *API Gateway Policy Developer Guide*.

Tip For recommendations on Ehcache security, see the following:

https://docs.oracle.com/javase/8/docs/technotes/guides/rmi/rmi_security_recommendations.html

Configure API Manager quota in multiple datacenters

API Manager quotas enable you to manage the maximum message traffic rate that can be sent by applications to APIs for back-end protection. You can use the **Server Settings > API Manager > Quota Settings** in Policy Studio to configure how API Manager quota information is stored. By default, quotas are stored in external storage, and automatically adapt to the configured KPS storage mechanism. However, you can also explicitly configure a storage mechanism of Cassandra, RDBMS, or in memory only.

API Manager quota storage

The following general guidelines apply to API Manager quota storage:

- Storing quota in memory only means that the quota calculation is performed by each API Gateway instance in each group between datacenters. If the quota duration is less than 30 seconds, in memory-only storage is automatically activated.
- It is best to use API Manager system quota (stored in Cassandra) when the back-end is shared between datacenters.
- It is best to use the API Gateway Throttling filter (stored in Ehcache) for back-end protection per datacenter.

For more details on configuring quotas in API Manager, see the *API Manager User Guide*.

Further details

For more details on using API Gateway environment variables in `envSettings.props`, see the *API Gateway DevOps Deployment Guide*.

For more details on Cassandra and Ehcache, see the following:

- http://ehcache.org/
- http://cassandra.apache.org/
- http://docs.datastax.com/en/cassandra/2.2/

For more details on configuring API Management in multiple datacenters, see:

- *Multi-datacenter deployment* on page 66
- *Multi-datacenter failover scenarios* on page 87

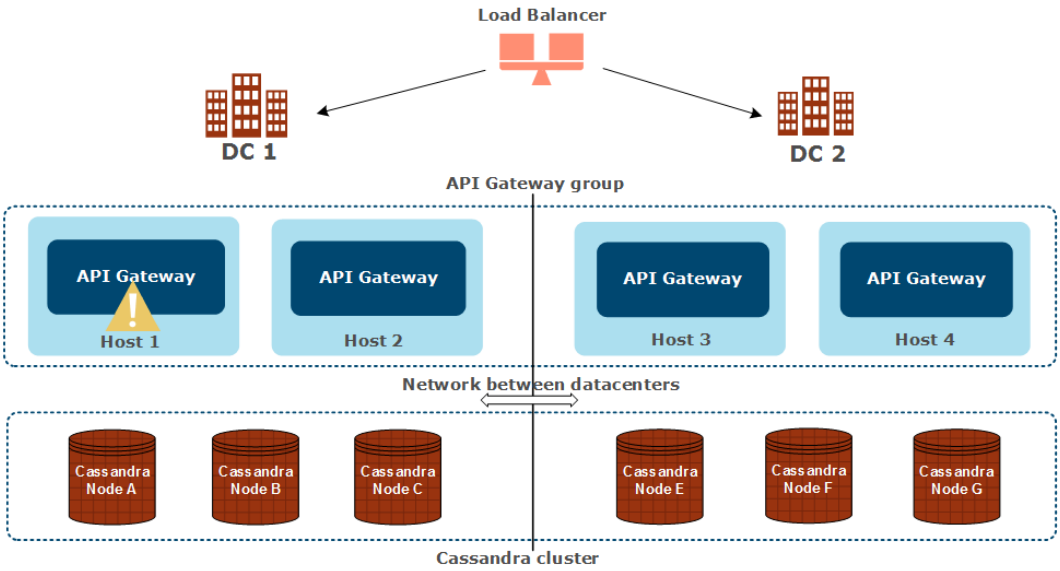
Multi-datacenter failover scenarios

This topic describes expected behavior in a multi-datacenter deployment in case of failover. It explains how the system will behave in each of the following scenarios:

- One API Gateway instance is down
- One Apache Cassandra node is down
- A full datacenter is down
- The network between two datacenters is down

One API Gateway instance is down

In this case, one API Gateway instance is down in DC 1:

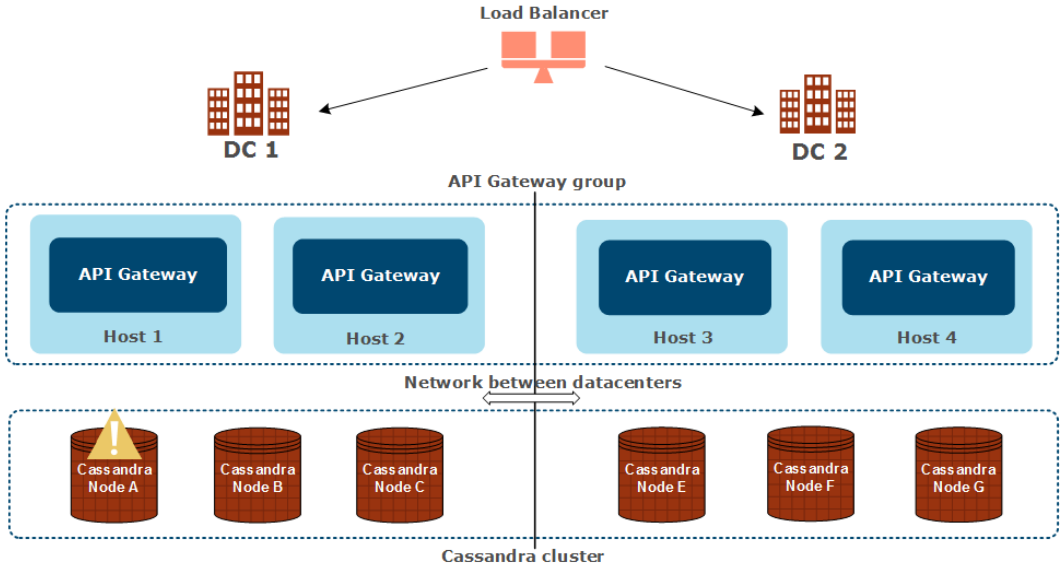


The following applies in this scenario:

- API requests can no longer be serviced by the API Gateway instance that is not running in DC 1.
- API requests will be serviced by the remaining API Gateway instance in DC 1.
- You must restart the API Gateway instance that is not running in DC 1. For more details, see *Start API Gateway* on page 40.

One Cassandra node is down

In this case, one Cassandra node is down in DC 1:



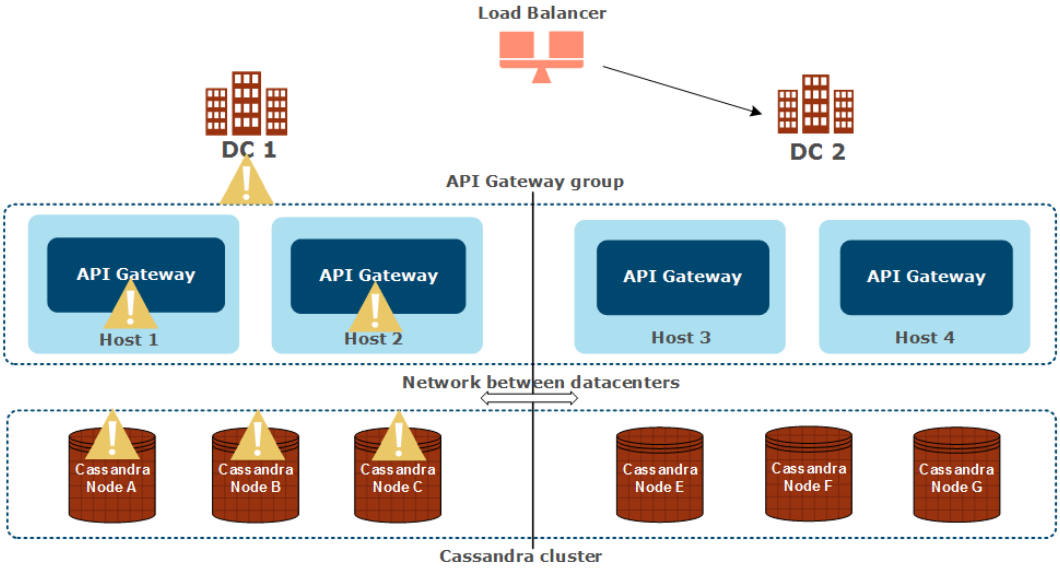
The following applies in this scenario:

- Cassandra is inherently HA and can tolerate the loss of one Cassandra node only in a datacenter (DC 1 in this case). This ensures 100% data consistency when Cassandra is configured for multiple datacenters. For more details, see *Configure Cassandra for multiple datacenters* on page 71
- You must restart the Cassandra node that is not running in DC 1. For details, see *Manage Apache Cassandra* in the *API Gateway Apache Cassandra Administrator Guide*.

Note When a node has been absent from a cluster for a time, it is brought back into the cluster after restart, and becomes eventually consistent by design. Node repair is required after re-integration into the cluster. For more details, see "Perform essential Cassandra operations" in the *API Gateway Apache Cassandra Administrator Guide*.

A full datacenter is down

In this case, DC 1 is down:



The following applies in this scenario:

- API requests can no longer be serviced by DC 1
- API requests are automatically directed to DC 2 by the load balancer
- API Manager quotas remain the same but over less servers
- You should not deploy updates for the following file-based data types to the API Gateway group:
    - API Gateway configuration
    - API Gateway KPS custom table structure

Note There is a risk that end-users may need to re-initiate sessions using the following data types stored in Ehcache:

- API Gateway OAuth token store
- API Gateway custom cache

Restart the datacenter

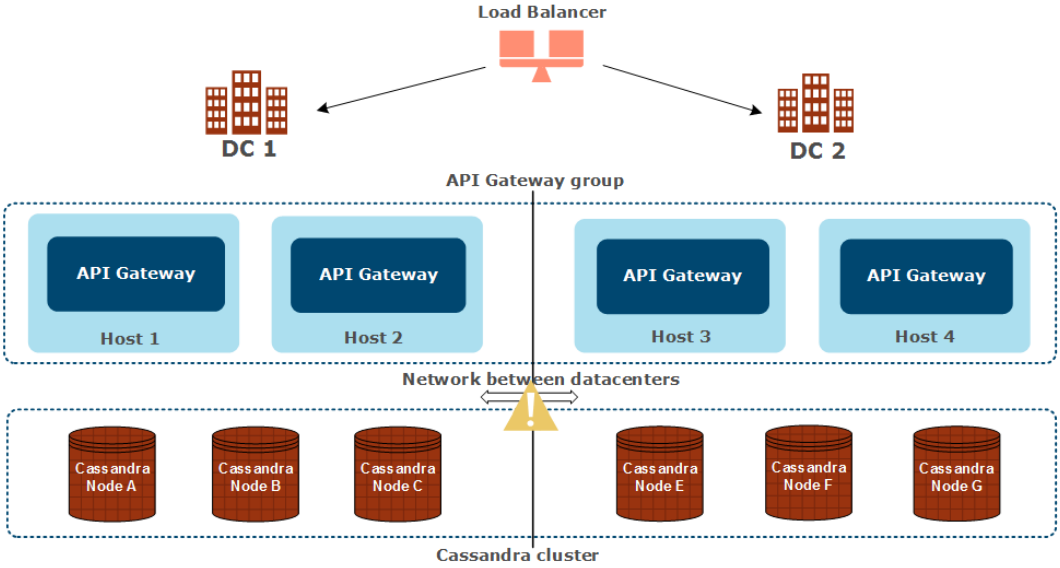
To restart the datacenter:

1. Restart each of the Cassandra nodes. For details, see *Manage Apache Cassandra* in the *API Gateway Apache Cassandra Administrator Guide*.
2. When all Cassandra nodes are running normally and have synchronized with the rest of the cluster in DC2, you can restart the API Gateway instances. For details, see *Start API Gateway* on page 40. API Gateway should not be restarted prior to a successful restart of all Cassandra nodes.

Note You should run the `nodetool repair` command on each of the three Cassandra nodes in DC 1 when the datacenter has been down for more than two hours.

The network between both datacenters is down

In this case, the network between DC 1 and DC 2 is down, while both datacenters remain active:



The following applies in this scenario:

- OAuth or throttling data stored in Ehcache per datacenter is not affected
- You should not deploy updates for the following file-based data types to the API Gateway group:
  - API Gateway configuration
  - API Gateway KPS custom table structure
- For the following data types stored in Cassandra, both datacenters are not synchronized until the network recovers, and then automatically resynchronize:
  - API Manager catalog, client registry, web-based settings
  - API Gateway KPS custom table structure
- For the following file based data types, a single API Gateway Manager web console cannot access both datacenters while the network is down, and can only access each datacenter separately:
  - API Gateway logs
  - API Gateway traffic monitoring

Note If the network connection has been down for more than two hours, the following steps are recommended:

- Run `nodetool repair` on each of the six nodes in the Cassandra cluster to ensure that the data has synchronized. For more details, see "Perform essential Cassandra operations" in the *API Gateway Apache Cassandra Administrator Guide*.

- Restart the API Gateway instances to resynchronize data from Cassandra (potentially in both datacenters if Cassandra changes have occurred in both datacenters).

It may take a minute for newly created, deleted, or updated APIs in one datacenter to synchronize successfully with the other datacenter.

Further details

For more details on how to configure API Management in multiple datacenters, see:

- *Multi-datacenter deployment* on page 66
- *Multi-datacenter configuration* on page 69

Update API Gateway 10

This section describes how to apply service packs or patches to API Gateway components.

Note Windows is supported only for a limited set of developer tools, see *Install developer tools on Windows* on page 60. API Gateway and API Manager do not support Windows.

Install a service pack or patch

This section describes how to install a service pack or patch on an existing installation of API Gateway.

To install a service pack or patch, follow these general guidelines:

1. Stop any Node Managers and API Gateway servers.
2. Back up your existing installation. For more information on backing up, see "API Gateway backup and disaster recovery" in the *API Gateway Administrator Guide*.
3. Download the service pack or patch and the associated *Readme* from Axway Support at https://support.axway.com.
4. Review the *Readme* for any specific installation instructions (for example, backing up any customized files used by API Manager or third-party tools).
5. Unzip and extract the service pack or patch. A service pack or patch contains new API Gateway binaries and does not overwrite the existing API Gateway configuration.
6. Restart the Node Managers and API Gateway servers.
7. To verify that the service pack or patch have been installed correctly, run the `managedomain --version` command.

   For more information on running the `managedomain --version` command, see "Get help with API Gateway" in the *API Gateway Administrator Guide*.

Resolve patch validation issues

You can use the `managedomain --version` command to list and validate the patches installed. This command uses the information in the in the `META-INF/<patch>.id` file to validate the patch. For more information on running the `managedomain --version` command, see "Get help with API Gateway" in the *API Gateway Administrator Guide*.

A patch that validates successfully is listed with no messages. If patch validation fails, a status message is displayed for each patch entry that failed to validate in the following format:

```
<status>: <patch_entry>: <message>
```

The possible message statuses are:

Status	Description
`Error`	An error has been detected for the `<patch_entry>`. You must take some action to fix the error.
`Info`	An informational message about the `<patch_entry>`. This does not usually require any corrective action.
`Warning`	A warning message about the `<patch_entry>`. Warnings indicate potential problems which might require you to take some action.

The `<patch_entry>` indicates the file or directory within the patch to which the message applies.

Some common messages, along with descriptions and suggested actions, are detailed in the following tables.

Cannot validate, no checksum available

Status	`Info`
Message	`Cannot validate, no checksum available`
Description	The `<patch_entry>` does not have a checksum value assigned in the `META-INF/<patch>.id` file.
Action	No action required if the `<patch_entry>` is a directory.

Content changed

Status	`Warning`
Message	`Content changed`
Description	The `<patch_entry>` checksum value differs from the one configured in the `META-INF/<patch>.id` file. This indicates that the file on disk has changed since the patch was installed.

Action	No action is required if the <patch_entry> indicates a configuration file that you have customized after patch installation. If the <patch_entry> does not indicate a configuration file that you have customized, check if there are two patches installed that patch the same file. You might be able to remove one of the patches (unless it also patches other files). If this is not the case, reinstall the patch.

File not found

Status	Error
Message	File not found
Description	An expected <patch_entry> cannot be found in the installation directory. This might indicate a partially removed patch, for example, a patched JAR file has been removed, but not the related META-INF/<patch>.id file.
Action	If you meant to delete the patch, remove the patch .id file to delete it completely. If you did not mean to delete the patch, reinstall it.

Malformed file

Status	Error
Message	Malformed file
Description	Either a <patch_entry> is misconfigured with more than one checksum value or the META-INF/<patch>.id file is malformed.
Action	Reinstall the patch, as the .id file might have been corrupted in some way.

Unexpected file

Status	N/A
Message	Unexpected file
Description	A <patch_entry> was found in the installation directory but is not expected based on the information in the META-INF/<patch>.id file. You might have removed a META-INF/<patch>.id file, but not the patch JAR files.

Action	Remove the JAR file. Alternatively, if you think you mistakenly deleted the `.id` file, reinstall the patch.

Verify which hosts have service packs or patches installed

In a multi-host environment, service packs and patches are installed on a host-by-host basis. In this scenario, you can use the API Gateway Manager web console to verify exactly which hosts have service packs or patches installed.

For example, you could upgrade host 1 to version 7.6.2 SP1, while host 2 remains at 7.6.2 for a period of testing. However, a system should run the same version across all hosts. You can use the API Gateway Manager topology and grid views to verify that all hosts in the system are running the same version and service pack. If a version mismatch is identified, you should ensure that the required service pack is installed on hosts that are running older versions.

For more details, see the *API Gateway Administrator Guide*.

License acknowledgments

Axway API Gateway uses several third-party toolkits to perform specific types of processing. In accordance with the Licensing Agreements for these toolkits, the relevant acknowledgments are listed below.

Acknowledgments

Apache Software Foundation:
This product includes software developed by the Apache Software Foundation.

OpenSSL Project:
This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit.

Eric Young:
This product includes cryptographic software written by Eric Young (eay@cryptsoft.com).

James Cooper:
This product includes software developed by James Cooper.

iconmonstr:
This product includes graphic icons developed by iconmonstr.