NetIQ Secure API Manager 1.1

Installation Guide

May 2020

Legal Notice

The only warranties for products and services of Micro Focus and its affiliates and licensors ("Micro Focus") are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Micro Focus shall not be liable for technical or editorial errors or omissions contained herein. The information contained herein is subject to change without notice.

For additional information, such as certification-related notices and trademarks, see http://www.microfocus.com/about/legal/.

Contents

About this Book

The *NetIQ Secure API Manager Installation Guide* provides conceptual information and step-by-step guidance for installation tasks.

Intended Audience

This guide provides information for individuals responsible for installing and maintaining Secure API Manager and connecting it to NetIQ Access Manager. You must have Access Manager installed and you must understand Access Manager, networking concepts, and virtual environments. NetIQ delivers the Secure API Manager as an appliance.

System Administrators

Deploy Secure API Manager across a distributed network. Configure Secure API Manager to work with Access Manager and configure virtual environments to run the Secure API Manager appliance.

Other Information in the Library

The library provides the following information resources in addition to this guide:

Release Notes

Provide information specific to this release of the Secure API Manager product, such as known issues.

NetIQ Secure API Manager Administration Guide

Provides details of configuration and administration tasks specific to this release of Secure API Manager.

NetIQ Secure API Manager API Management Guide

Provides detailed information about how to add APIs to a central repository, manage the APIs, and maintain the APIs throughout their lifecycle.

1 Secure API Manager Overview

Application programming interfaces (APIs) are sets of definitions, protocols, and tools for building software. Much software and many items that make up the Internet of Things (IoT) use APIs to provide functionality that your business requires. The APIs also provide the ability to customize software to solve your business problems.

Secure API Manager gives you a single place to add, manage, audit, and secure the APIs that your company uses. You add the APIs once to Secure API Manager and they are available for reuse. You can see all of the available APIs in a single location, making it easy for you to combine multiple APIs to create new functionality while seamlessly requiring access to the APIs through NetIQ Access Manager.

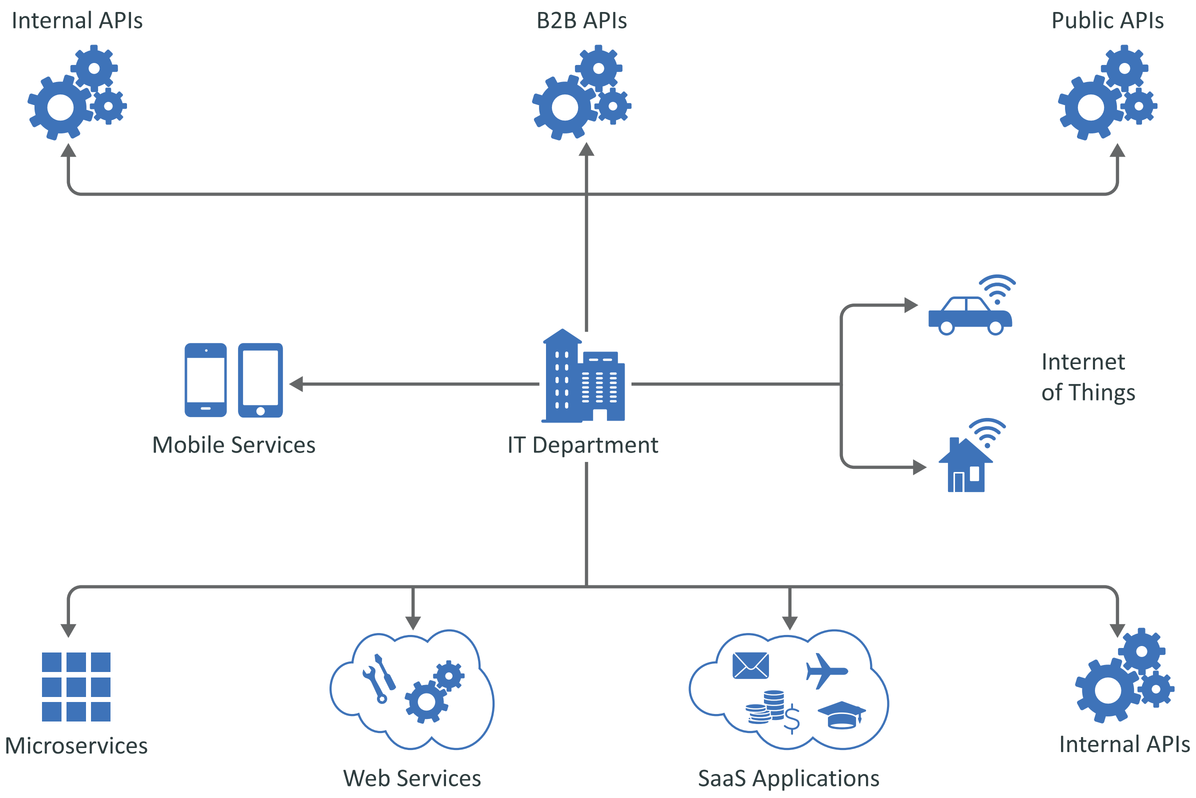
How Secure API Manager Solves API Management Issues

The use of APIs in IT environments has grown significantly in corporate IT environments and many businesses now build their own APIs to develop new services for their users. APIs can be built and implemented more quickly - and provide more flexibility and scalability - than traditional offerings.

Customers often begin to explore API management solutions when they are considering an application transformation project. For example, you might want to achieve mobile integration, create an API-enabled hybrid infrastructure (such as cloud and on-premise workloads, microservices, and so on), or even implement a complete digital transformation project across your environment.

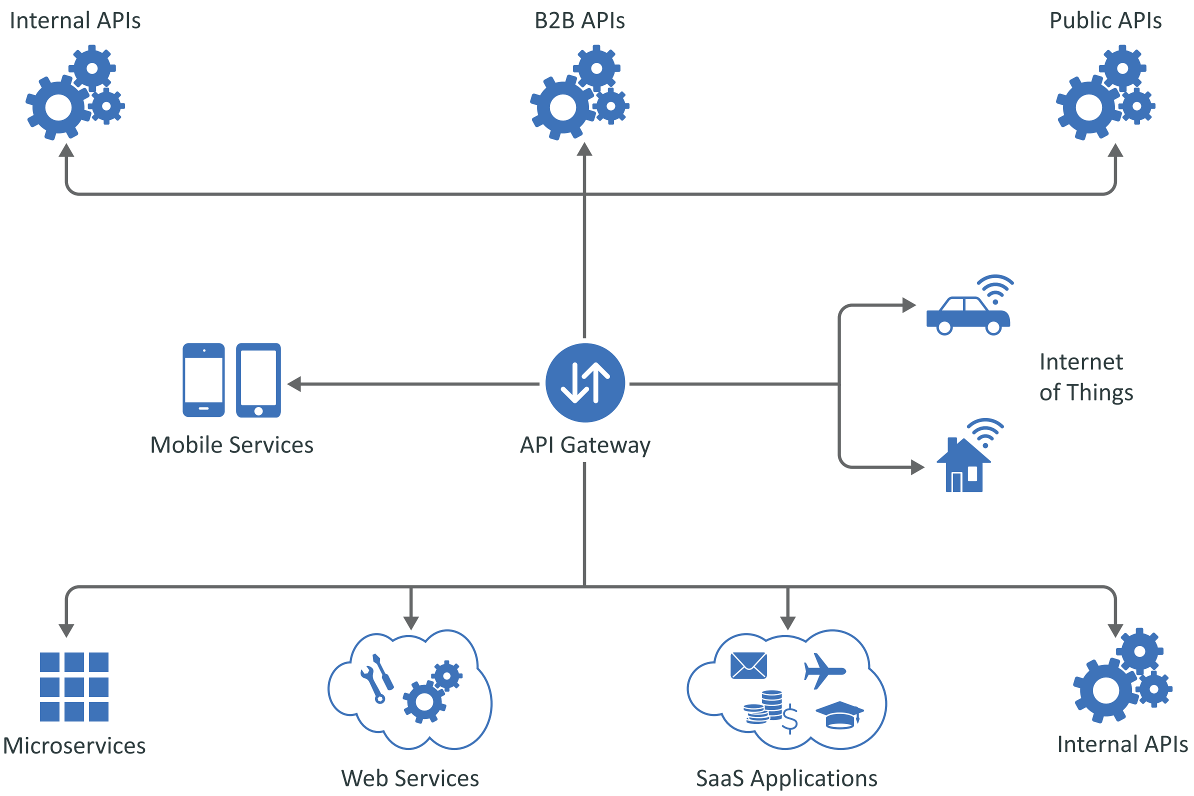
As attractive as the use of APIs might be, there is no doubt that managing hundreds or even thousands of public APIs, internal APIs, or business-to-business APIs across mobile devices, web services, public and private clouds, microservices, and so forth, can become very complex. The following graphic depicts how a company can use APIs in its IT environment.

Figure 1-1 How Companies Use APIs



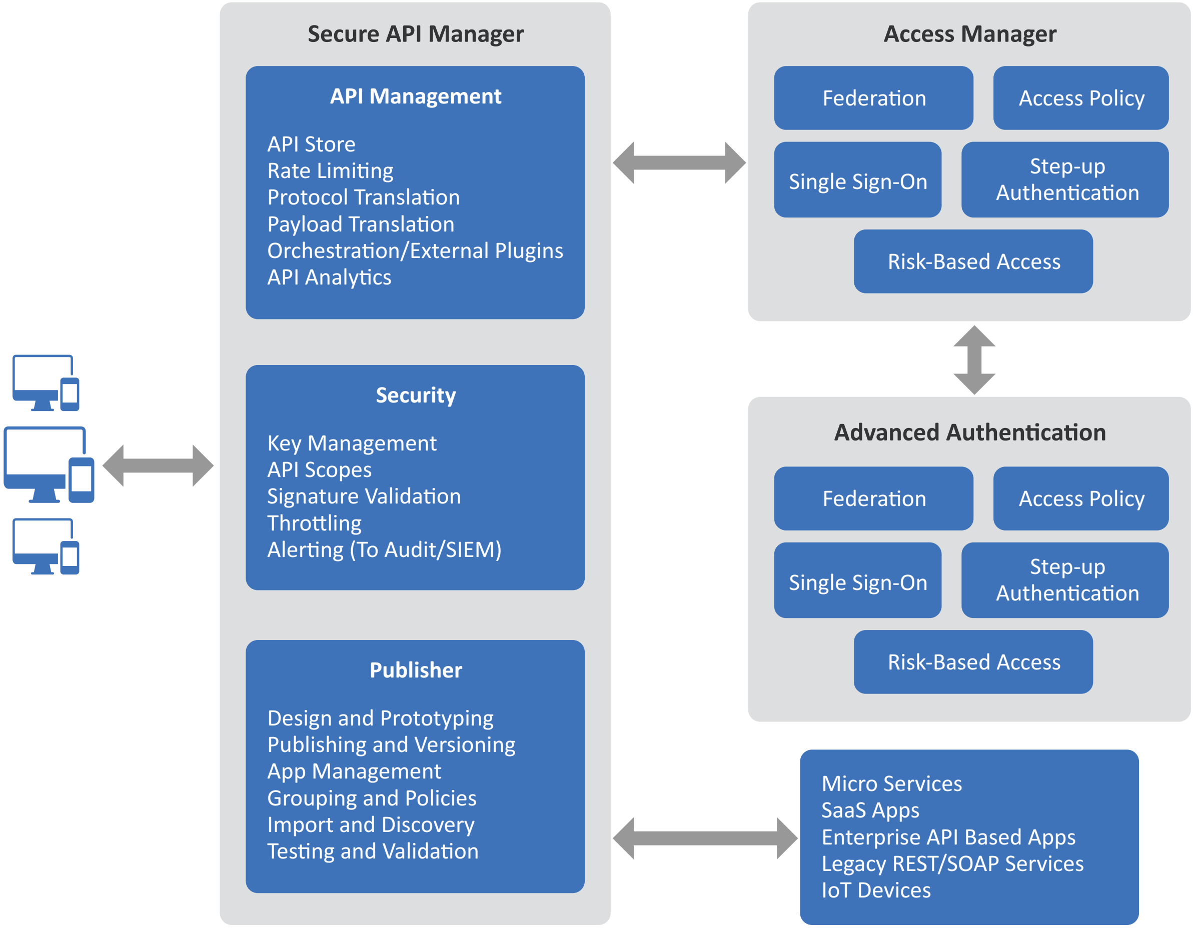
Regardless of your goals, Secure API Manager can solve many of the issues associated with API management. It enables you to manage, create, control, and audit the APIs used in your environment. It provides an API Gateway that manages all the API traffic in your company.

Figure 1-2 How Secure API Manager Controls APIs



Secure API Manager is a solution that you add to Access Manager. You must have Access Manager installed and running before you can deploy Secure API Manager. The following graphic depicts the solutions that NetIQ provides when you combine different products together.

Figure 1-3 *Secure API Manager Solution*



Secure API Manager provides the following solutions for managing APIs:

- A single repository for all of your APIs
- Secure access to the APIs because of the integration with NetIQ Access Manager
- A lifecycle system to track the state of the APIs
- Throttling capabilities to limit throughput to certain APIs
- A detailed analytics system to show you which APIs are being used the most

The purpose of this guide is to help you understand how to use Secure API Manager to add, manage, and secure the APIs for your company.

Understanding the Secure API Manager Components

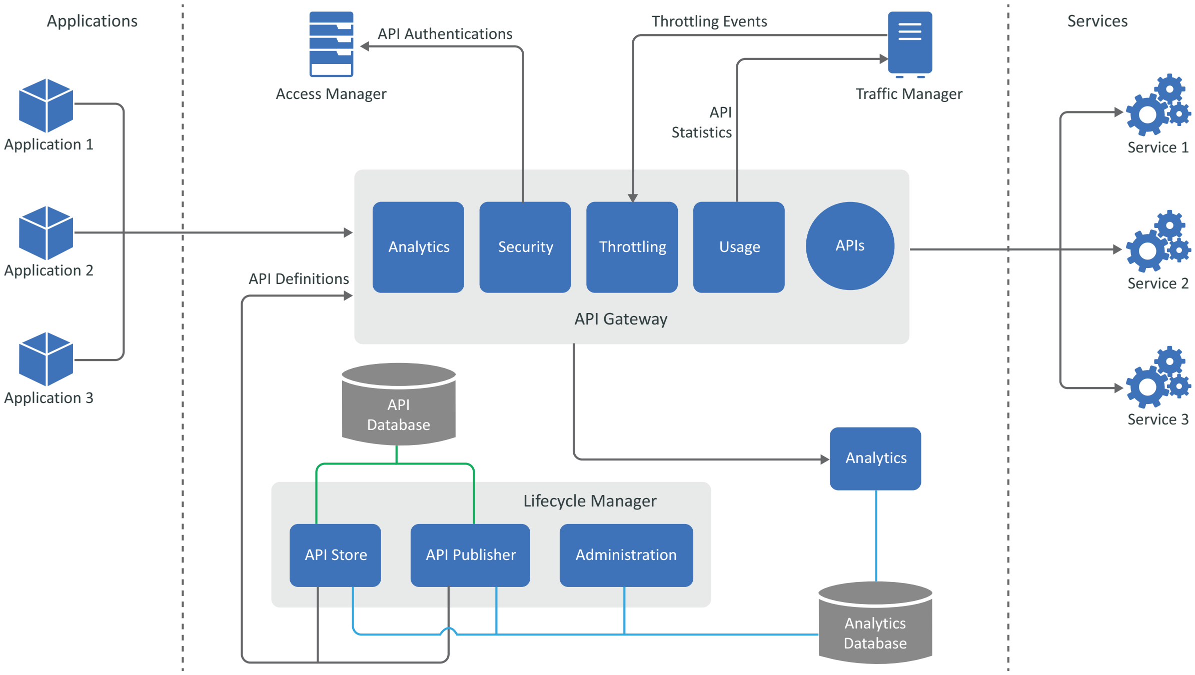
NetIQ provides Secure API Manager as an appliance that you can deploy in your existing virtual (VMware) environment. Secure API Manager has four separate components: Analytics, API Gateway, Database Service, and Lifecycle Manager.

Secure API Manager allows you to deploy the components in different configurations depending on your environment. For more information, see "Understanding Deployment Scenarios" on page 19.

IMPORTANT: Running all of the components on one virtual machine is not supported in a production environment. Deploying all components on one virtual machine is supported only for testing purposes.

The following graphic provides a high-level architectural view of Secure API Manager. The graphic shows the different components of Secure API Manager and how they interact with each other. One important thing to note is that Secure API Manager requires Access Manager to work.

Figure 1-4 *Secure API Manager Architecture*



The following sections provide details about the different components of Secure API Manager.

- “Analytics” on page 11
- “API Gateway” on page 12
- “Database Service” on page 12
- “Lifecycle Manager” on page 12

IMPORTANT: We recommend as a best practice that you deploy each component on its own appliance in a production environment. You *must* run the Database Service, API Gateway, and Lifecycle Manager on separate appliances. You should also install the Analytics component on its own appliance if you plan to run a lot of reports.

Analytics

The Analytics component provides detailed logs about the number of authorizations to each API, which APIs have been combined to create applications, and where the authorizations are coming from, among many other items. There are no configuration options for the Analytics component. The Analytics reports work as long as you deploy the Analytics component.

API Gateway

Integrating Secure API Manager with NetIQ Access Manager ensures that only the approved calls are made to the APIs through OAuth tokens. The API Gateway component controls the number of authorizations to the API through the use of throttling policies. All API communications go through the API Gateway to create audit trails and to provide detailed analytics about each API.

Database Service

Many of the services in Secure API Manager require a database to function. The Database Service component provides multiple databases for different services. For example, there is a database that contains all of the APIs and a database for analytics. When you deploy the Database Service, it deploys the required databases for Secure API Manager to work.

WARNING: The Database Service component must run on its own appliance. Do not combine any other components with the Database Service component.

Lifecycle Manager

The Lifecycle Manager component consists of the consoles responsible for creating, testing, managing, and deprecating APIs. It also contains the administration consoles for Secure API Manager and Analytics. These different components are:

- **Publisher:** The Publisher is where you add the APIs to the single repository. You can see all available APIs in one location and view the analytics of the APIs in this console. You access the Publisher at `https://`*`lifecycle-manager-dns-name`*`:9444/publisher`. For more information, see "Accessing the Publisher" in the *NetIQ Secure API Manager 1.1 API Management Guide*.
- **Store:** The Store displays all available APIs to the developers who want to use the APIs. The Store also allows developers to combine two or more APIs together to create applications. You access the Store at `https://`*`lifecycle-manager-dns-name`*`:9444/store`. For more information, see "Accessing the Store" in the *NetIQ Secure API Manager 1.1 API Management Guide*.
- **Management console:** The management console allows you to configure roles, view logs, and manage other aspects of Secure API Manager. You access the console at `https://`*`lifecycle-manager-dns-name`*`:9444/carbon`. For more information, see "Accessing the Management Console" in the *NetIQ Secure API Manager 1.1 Administration Guide*.
- **Administration console:** The administration console allows you to create and manage groups as well as configure policies for throttling access to APIs. You access the console at `https://`*`lifecycle-manager-dns-name`*`:9444/admin.` For more information, see "Accessing the Administration Console" in the *NetIQ Secure API Manager 1.1 Administration Guide*.

You can deploy all components on one appliance for testing purposes. You *must* run the Database Service, API Gateway, and Lifecycle Manager on separate appliances. We recommend as a best practice in an enterprise environment that you deploy each component on a separate appliance. For more information, see "Understanding Deployment Scenarios" on page 19.

There is an appliance management console available for each appliance that you deploy. The appliance management console allows you to manage that specific appliance. For example, if you cluster the appliance for load balancing and high availability, the appliance management console allows you to apply patches to each appliance in the cluster. For more information, see “Managing the Appliance” in the *NetIQ Secure API Manager 1.1 Administration Guide*.

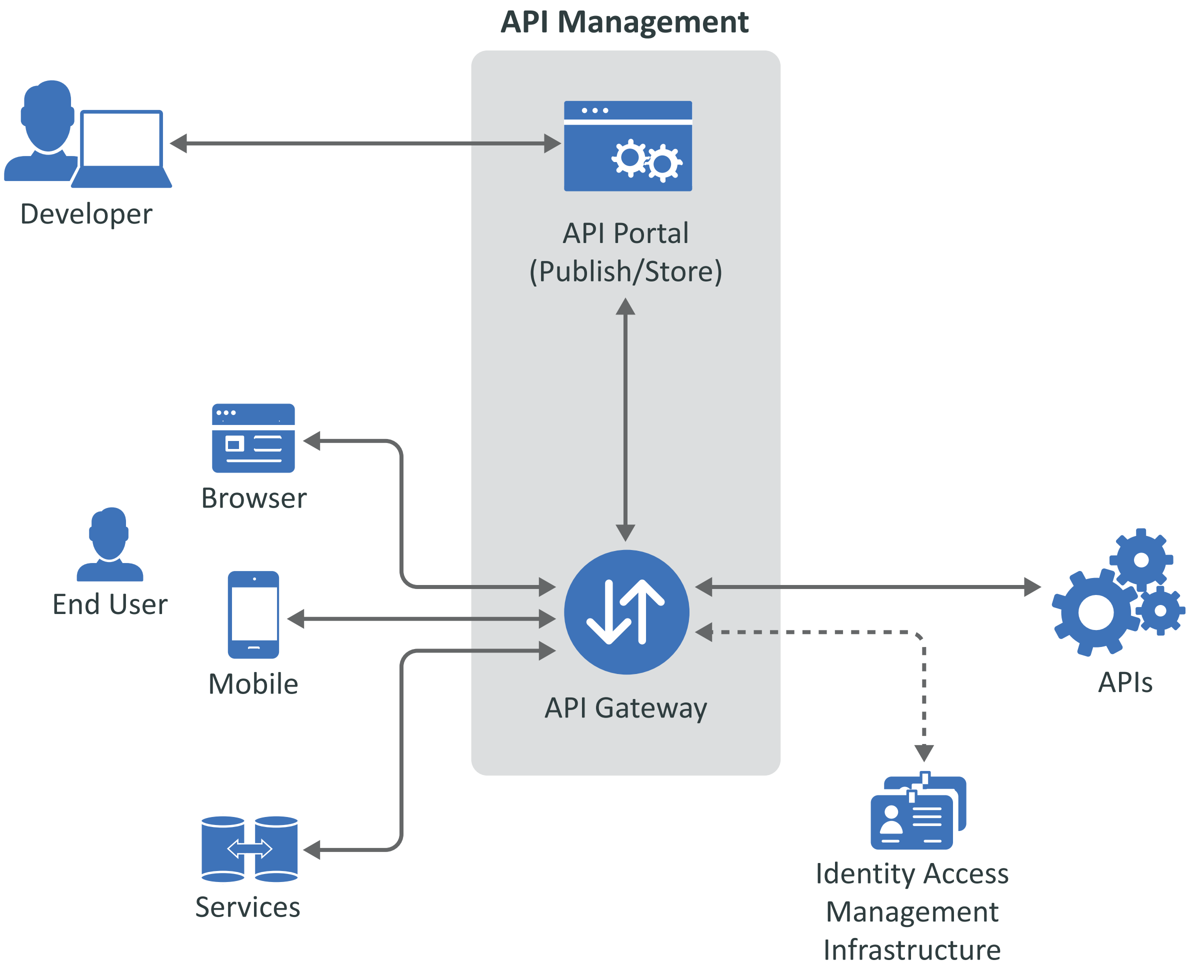
How Secure API Manager Works

There are two points of integration between Secure API Manager and Access Manager. The following graphic depicts how Access Manager ensures that all requests that come to the API Gateway are authorized requests through the use of OAuth2 tokens.

The second integration point allows you to control who has access to which APIs through the use of the Access Manager scopes and roles. For more information, see "Understanding How Secure API Manager Uses the Access Manager Scopes and Roles to Determine API Access" in the *NetIQ Secure API Manager 1.1 API Management Guide*.

The following graphic depicts the management of the APIs through Secure API Manager.

Figure 1-5 *API Management in Secure API Manager*



1. Developers add or create the APIs through the Publisher and combine and use the available APIs in the Store.
2. End users execute an application or service through a browser or a mobile device. The application or service makes a call to the API stored in the API Gateway component of Secure API Manager.

3. The API Gateway component of Secure API Manager takes the API request and sends a request to the OAuth application in Access Manager, that you create during the configuration phase, for an authorization token for the API. Access Manager ensures that the API request is a valid request and issues a token for authorization of the API.
4. The API Gateway receives the approval or denial for the API authorization request from Access Manager and then allows the API in the API Gateway to execute or deny access to the API. The execution of the API provides an additional feature or function to the application or service that the end user is running.

This process ensures that Secure API Manager accepts only valid requests. This type of access control ensures that no denial of service attacks can take down the system. The second level of access control allows Secure API Manager to integrate with Access Manager to use the Access Manager scopes and roles to limit who has access to which API or API endpoint. For more information, see "Understanding How Secure API Manager Uses the Access Manager Scopes and Roles to Determine API Access" in the *NetIQ Secure API Manager 1.1 API Management Guide*.

2 Planning to Install Secure API Manager

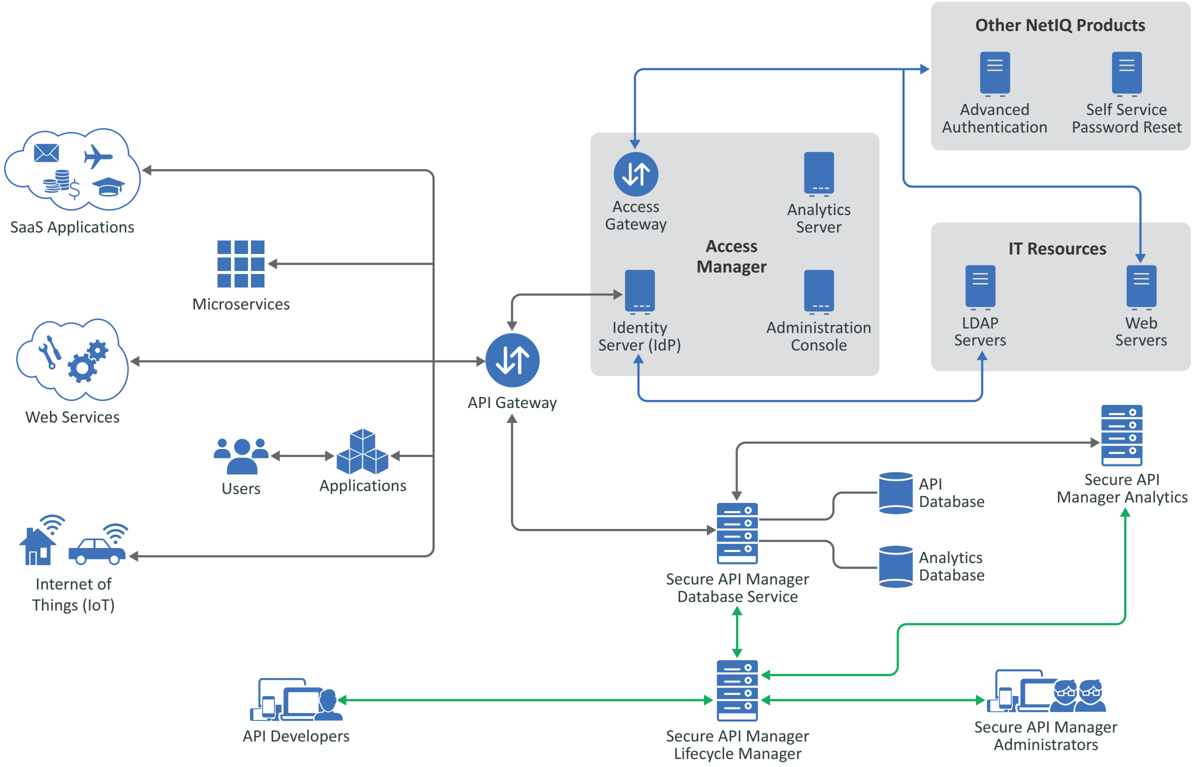
Secure API Manager requires that you have a deployment of Access Manager up and running before deploying Secure API Manager. If you do not, the deployment of Secure API Manager will fail. Use the following section to plan a successful deployment and configuration of Secure API Manager in your IT environment.

Understanding the Flow of Communications through Secure API Manager

The Secure API Manager components communicate securely through SSL. This means that you must have a trusted root certificate or use the self-signed certificate on the appliance to deploy Secure API Manager. Secure API Manager does not allow non-SSL communication between the different components.

Secure API Manager uses Access Manager to create OAuth tokens that allow secure access to the APIs. The following graphic depicts the flow of information between Secure API Manager and Access Manager.

Figure 2-1 *Secure API Manager Communication Flow*



API developers create and add APIs to the API Gateway through the Lifecycle Manager. The developers must have access to the Lifecycle Manager. The Lifecycle Manager provides the ability to test the APIs, maintain a lifecycle of the APIs, and control the number of authorizations to the APIs through the throttling policies.

After the developers create or add the APIs to the API Gateway, the flow of communication occurs in the following manner:

1. The application or service makes a call to the APIs stored in the API Gateway.

2. The API Gateway contacts the Identity Provider in Access Manager to obtain the OAuth token to ensure that the application or service is approved to make the call to the APIs.
3. The Identity Provider validates the request and sends an OAuth token back to the API Gateway. The API Gateway then uses that token to make the authorized API calls to provide the additional functionality to the service or application through the APIs. For more information, see “Configuring Secure API Manager” in the *NetIQ Secure API Manager 1.1 Administration Guide*.

You must ensure that the applications and services can communicate with and receive information from the API Gateway. You must also ensure that the API Gateway can communicate with and receive information from the Identity Provider in Access Manager.

Understanding Deployment Scenarios

Secure API Manager has four components: Analytics, API Gateway, Database Service, and Lifecycle Manager. Each component performs a different function for Secure API Manager. For more information, see “Understanding the Secure API Manager Components” on page 10. In a test environment, you can deploy all components on one appliance. In a production environment, there are some restrictions and limitations. Use the following information to plan your deployment configuration.

- “Deployment Considerations and Restrictions” on page 19
- “Testing Deployment Scenario” on page 20
- “Enterprise Deployment Scenario On-Premises” on page 20
- “Enterprise Deployment Scenario in AWS” on page 21

Deployment Considerations and Restrictions

Determining how to deploy the components depends on many different variables:

- Network environment
- Number of APIs stored in the API Gateway
- Number of API calls
- Number of people adding APIs and creating applications
- Analytics usage
- Location of Deployment

IMPORTANT: We recommend as a best practice that you deploy each component on its own appliance in a production environment.

If you do not deploy each component on its own appliance, you must still adhere to the following requirements for deploying the different components.

- **Database Service:** The Database Service component must run on its own appliance. Do not combine any other components with the Database Service component. The Database Service component keeps track of configuration information and user accounts. Running other components with the Database Service can cause corruption of the configuration files.

- **Lifecycle Manager and API Gateway on separate appliances:** To ensure data integrity, you must deploy the Lifecycle Manager and the API Gateway on separate appliances. You must use the same NFS server but you must define and use separate mount points.

  IMPORTANT: Once you have installed the Lifecycle Manager and API Gateway components on separate appliances, if you want to deploy additional Lifecycle Manager and API Gateway components in your environment at a later time, you must again deploy them on separate appliances. Attempting to use different configurations of the Lifecycle Manager and the API Gateway will result in database corruption on the NFS mount point.

The Analytics and Database Service components use a lot of disk space and processing power. Running the Analytics component on its own appliance greatly increases the performance of the overall system.

Testing Deployment Scenario

You can deploy all four components on one appliance but this configuration is only for testing purposes. Running all of the components on one appliance drastically reduces the performance of the entire Secure API Manager system. You cannot cluster a test system. You can run a test system on-premises or in Amazon Web Services (AWS).

IMPORTANT: Deploying all four components on one appliance is supported *only* for testing purposes. It is not supported in a production environment.

Enterprise Deployment Scenario On-Premises

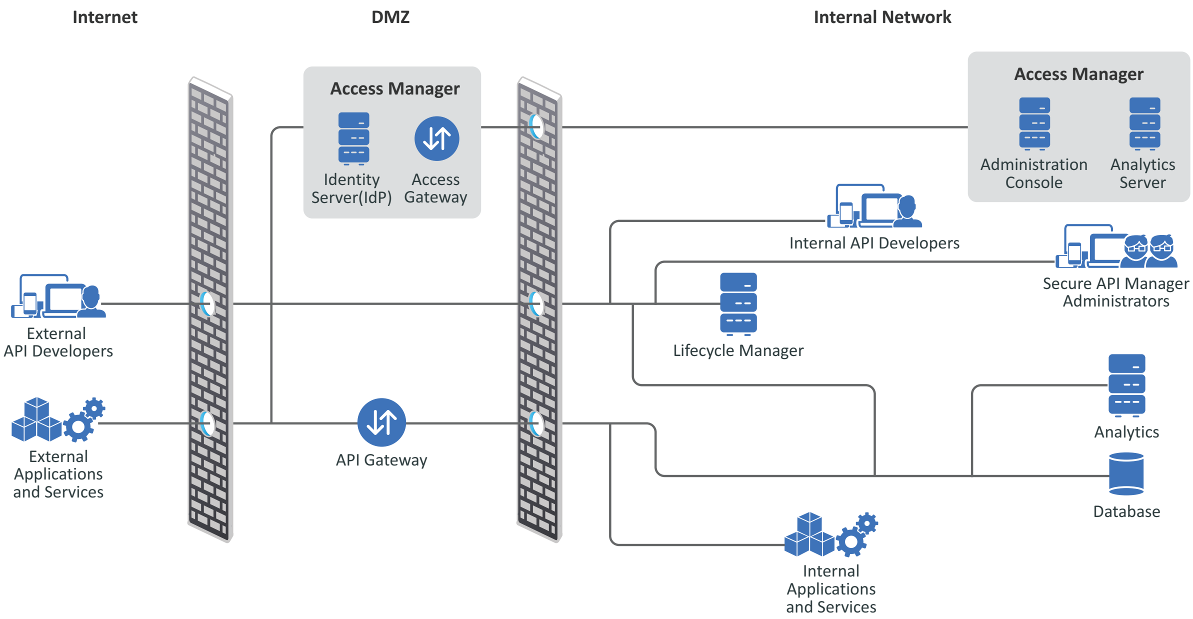
The following on-premises deployment scenario provides guidance on how to deploy the different components of Secure API Manager and where we recommend that you deploy those components. For enterprise environments, we recommend that you deploy each component on a separate appliance and that you cluster each component for load balancing and high availability. For more information, see “Enabling High Availability and Load Balancing” on page 23.

To cluster components, use an L4 switch. Clustering provides redundancy, high availability, and load balancing. We also recommend that you place the L4 switch for the API Gateway and Lifecycle Manager in the DMZ to allow external applications, services, and API developers access to Secure API Manager. You must ensure that the API Gateway component or the L4 switch for the API Gateway component can communicate with the Identity Provider in Access Manager. You must also ensure that API developers can communicate with the Lifecycle Manager.

WARNING: All components must have direct access to the primary database without going through an L4 switch or database corrupt can occur.

The following graphic depicts the recommended on-premises deployment scenario for enterprise environments. In this scenario, all of the components are deployed on separate appliances.

Figure 2-2 *Enterprise Secure API Manager Deployment On-Premises*



The appliances are clustered using an L4 switch for high availability and load balancing. The L4 switches for the API Gateway and the Lifecycle Manager are in the DMZ to allow external applications, services, and API developers access to Secure API Manager. The L4 switch for the API Gateway component can communicate with the Identity Provider in Access Manager, and API developers can also communicate with the Lifecycle Manager.

Enterprise Deployment Scenario in AWS

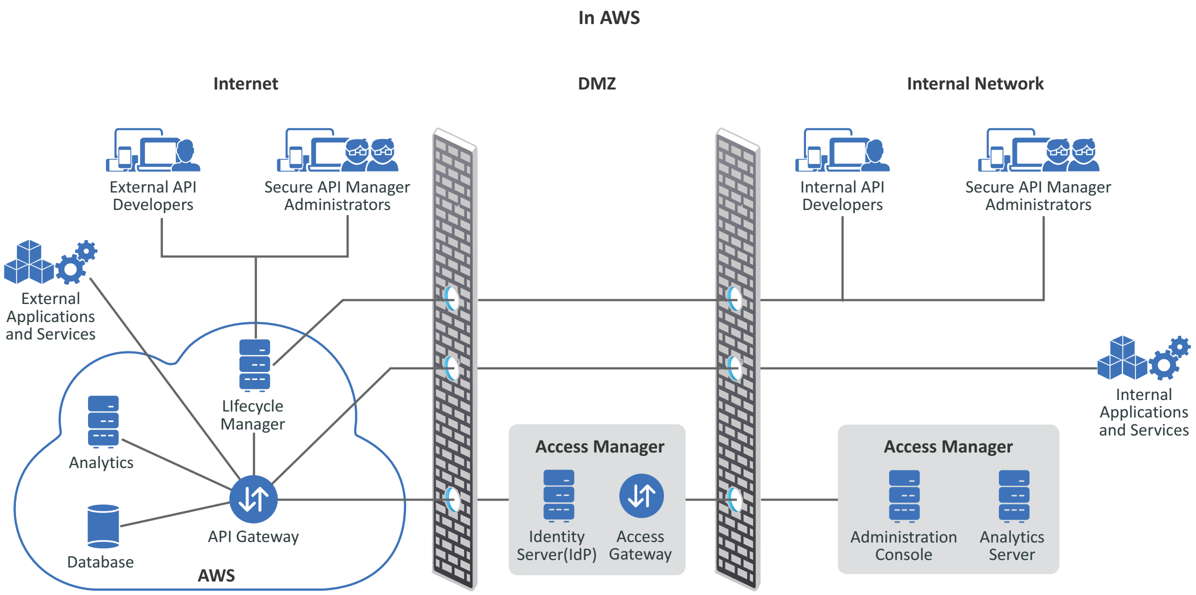
The following Amazon Web Services (AWS) deployment scenario provides guidance on how to deploy the different components of Secure API Manager and where we recommend that you deploy those components. For enterprise environments, we recommend that you deploy each component on a separate appliance and that you cluster each component for load balancing and high availability. For more information, see “Enabling High Availability and Load Balancing” on page 23.

To cluster components, use an L4 switch. Clustering provides redundancy, high availability, and load balancing. You must ensure that the API Gateway component or the L4 switch for the API Gateway component can communicate with the Identity Provider in Access Manager. You must also ensure that API developers can communicate with the Lifecycle Manager.

WARNING: All components must have direct access to the primary database without going through an L4 switch or database corrupt can occur.

The following graphic depicts the AWS deployment scenario for an enterprise environment. In this scenario, all of the components are deployed on separate appliances.

Figure 2-3 *AWS Enterprise Deployment for Secure API Manager*



Secure API Manager must have access to the Access Manager Identity Server in order to work properly. You must ensure that the API Gateway and the Identity Server can communicate with each other.

Using High Availability and Load Balancing with Secure API Manager

Secure API Manager supports high availability and load balancing for the different components with the use of an L4 switch. You must install and deploy an L4 switch for each component that you want to cluster in your environment, and also ensure that you use session persistence in the L4 switch. You must also install a Networking File System (NFS) server to provide content synchronization between the nodes in the clusters.

Secure API Manager uses your browser's session storage to facilitate seamless high availability and load balancing for the different Secure API Manager components. Here are some reasons why you would want to cluster the different components:

- **Analytics:** Clustering the Analytics server provides a backup of the analytics information in case of disasters or hardware failures.
- **API Gateway:** Clustering the API Gateway facilitates the API authorizations by load balancing the authorizations to the different nodes in the cluster and providing a backup of the APIs in case of a disaster or hardware failure.
- **Database Service:** Clustering the Database Service provides a backup of your configuration information stored on the Database Service in case of a disaster or hardware failure.

- **Lifecycle Manager:** Clustering the Lifecycle Manager allows for high availability to the Store, Publisher, management console, and administration console. If one node goes down, users can still access and use whichever console they need.

Use the following information to enable load balancing with an L4 switch and an NFS server.

- "Enabling High Availability and Load Balancing" on page 23
- "Configuring Content Synchronization for High Availability Using a Network File System Server" on page 24

Enabling High Availability and Load Balancing

Secure API Manager supports high availability and load balancing for the different components with the use of an L4 switch. If you want high availability and load balancing, you must install and deploy an L4 switch for each component that you want to cluster. If you use an L4 switch, ensure that you use session persistence in the L4 switch. For example, if you want to cluster the Database Service and Analytics, you must deploy two separate L4 switches. You deploy one L4 switch for each component you deploy.

Ensure that you use sticky sessions or session persistence in the L4 switches. Otherwise, as API developers and administrators are working and their existing sessions change, Secure API Manager requires users to re-authenticate before they can continue their work. If the L4 switches have sticky sessions, users and administrators do not have to re-authenticate.

You must configure the following ports in the L4 switch for the various Secure API Manager components.

Table 2-1 *L4 Ports*

Component	Ports
Database Service	5432: Postgres
Lifecycle Manager	9444: HTTPS Servlet Transport
	5673: Internal Message Broker
	9763: WebSockets
API Gateway	8246: NIO/PT Transport
	9446: HTTPS Servlet Transport
	9102: WebSockets
Analytics	7613: Throttling authentication
	7713: Throttling secure transport
	9443: HTTPS servlet transport

Use the following information to help you deploy an L4 switch for the components you want to cluster.

1 Install an L4 switch and ensure that you use session persistence.

2 Deploy two or more separate appliances for each component that you want to cluster. Each appliance must have a separate IP address and the L4 switch provides the DNS name of the component.

3 Ensure that the L4 switch is configured to use sticky sessions. For a given browser session, the session must remain on the same Secure API Manager node over time unless the Secure API Manager node becomes unavailable.

4 Follow the L4 switch documentation to configure the L4 switch to provide load balancing for the Secure API Manager nodes. Use the information provided in Table 2-1 to configure the appropriate ports on the L4 switch.

5 Repeat the steps for each component that you deploy.

Configuring Content Synchronization for High Availability Using a Network File System Server

Secure API Manager supports high availability and load balancing with an L4 switch in front of the different components to cluster the components. You must deploy at least two separate appliances of the same component for high availability and load balancing. However, you must also use a Network File System (NFS) server to synchronize the content between the multiple nodes in the cluster to provide high availability and load balancing.

Secure API Manager stores configuration information in the Database Service component as well as in the file system on the other components. When you cluster the Lifecycle Manager and the API Gateway, Secure API Manager uses the NFS server to synchronize the configuration files between the clustered nodes. If you do not use the NFS server, the files are not synchronized and you can lose the configuration information of your APIs and corrupt the APIs.

Secure API Manager mounts the *API-M_HOME*/repository/*deployment/server* directory from the two nodes to the shared file system, in order to share all APIs and throttling policies between all the nodes, thereby avoiding the vulnerability of a single point of failure.

Before you deploy Secure API Manager you must already have an NFS server in your IT environment or you must install and configure an NFS server. The Deployment Manager validates that the NFS server is up and communicating and if it is not, the deployment does not continue.

Secure API Manager does not provide an NFS server for you. You are responsible for maintaining the NFS server. You must perform the following steps to ensure that the NFS server contains the proper content for Secure API Manager to function.

1 Ensure that you have a supported version of the NFS server deployed and running in your IT environment. For more information, see “Deployment Requirements of Secure API Manager” on page 25.

2 Create an empty folder with the proper permissions on the NFS server where Secure API Manager stores the shared content.

2a Create an empty directory with any name. Ensure that you record this directory name for later use. For example: */sapim-gw*

2b Ensure that the directory has the correct NFS permissions (read, write, and execute).

2c For security, map the incoming Linux identities from Secure API Manager of user 802 group 802 by adding an entry to `/etc/exports` on the NFS server. For example:

```
/sapim-gw
*(rw,sync,no_subtree_check,all_squash,anonuid=802,anongid=802
```

2d Export this using the command `exportfs -a`.

3. During the configuration of the API Gateway and the Lifecycle Manager, you must enter the following information for the NFS server:

   sharedStorageHost

   Specify the IP address or DNS name of the NFS server.

   sharedStorageMount

   Specify the name of the directory you created on the NFS server in Step 2.

Obtaining Secure API Manager

You must have purchased Secure API Manager to access the product in the Customer Center. For more information, see How to Buy. The activation code for Secure API Manager is in the Customer Center where you download the software. If you have issues finding or accessing the activation code, see Customer Center Frequently Asked Questions.

The activation code allows you to receive security and product updates. If you do not enter the activation code, you do not receive updates. For more information, see Performing an Online Update.

To obtain Secure API Manager:

1. Log in to the Customer Center.
2. Click **Software**.
3. On the **Entitled Software** tab, click the appropriate version of Secure API Manager for your environment to download the product.
4. Download the product file and the ZIP file for email notifications.

Deployment Requirements of Secure API Manager

Secure API Manager provides a single image that allows you to choose which components to deploy on an appliance. You can deploy one, two, three, or all four components on one appliance depending on your environment. Ensure that you have read and understand how you should deploy Secure API Manager in your environment. For more information, see "Understanding Deployment Scenarios" on page 19.

The following table contains the minimum requirements required to deploy a single Secure API Manager appliance. Whether you deploy one or more components on an appliance, the requirements do not change.

NOTE: Deploying all of the components on one appliance is supported only for testing purposes. This configuration is not supported in a production environment.

Ensure that you meet these minimum requirements before deploying the product.

Table 2-2 *Secure API Manager Appliance Requirements*

Component	Requirements
Virtual system	VMware ESX 6.5 or later **NOTE:** Your VMware license must be Enterprise or Enterprise Plus if you want to use remote serial connections. For more information, see the VMware documentation (https://www.vmware.com/support/pubs/).
Hard disk space	60 GB (per appliance)
Memory	12 GB of RAM (per appliance)
Processors	4 (per appliance)
Browsers	◆ Google Chrome latest version ◆ Microsoft Edge latest version ◆ Microsoft Internet Explorer 11 with latest patches ◆ Mozilla Firefox latest version
Publicly Resolvable DNS Name	Each DNS name of the appliance must be publicly resolvable. Secure API Manager uses Docker containers to create the components. The DNS name of each appliance must be publicly resolvable to allow the Docker containers to access the local `/etc/hosts` file of the appliance. If the DNS name is not publicly resolvable, the components cannot communicate with each other and the product does not work.
IP Ports	Ensure that the default ports for Secure API Manager are open in your firewall. For more information, see "Ports for Secure API Manager" on page 27.
Trusted root certificate or self-signed certificate	The Secure API Manager components communicate securely over SSL. You must have a trusted root certificate or use a self-signed certificate to have the Deployment Manager work.
License	The license is required to receive online updates. Obtain the license from the Customer Center. You add the license to each appliance after you complete the installation. For more information, see "Performing an Online Update" in the *NetIQ Secure API Manager 1.1 Administration Guide*.
NetIQ Access Manager 4.5 or later	Secure API Manager is an add-on product for Access Manager 4.5 or later. You must have Access Manager deployed and running before deploying Secure API Manager. For more information, see Chapter 3, "Integrating Secure API Manager with Access Manager," on page 29.
Network File System (NFS) v3	If you cluster the component for high availability and load balancing, you must have a Network File System (NFS) server deployed and running in your IT environment that Secure API Manager uses. For more information, see "Using High Availability and Load Balancing with Secure API Manager" on page 22.

Ports for Secure API Manager

Secure API Manager uses various ports to communicate with Access Manager, the databases, the different components, and NetIQ so that the appliances can receive patches and upgrades. Your deployment determines which ports the appliances use. You can view the open ports through the appliance management console. For more information, see "Viewing the Open Ports in the Firewall" in the *NetIQ Secure API Manager 1.1 Administration Guide*.

WARNING: Do not change any of the firewall settings on the appliances that you deploy. Secure API Manager automatically configures the firewall setting on each appliance for you. If you do change the firewall settings on the appliances, the Secure API Manager system is no longer supported.

Use the following information to help you properly configure your firewalls external to the appliances. The table below is not complete. The following items are some of the most common ports the appliances use. Ensure that you do not block the ports, otherwise you might disable communication between the components or it might cause you not to receive patch updates and upgrades.

Ensure that you understand the communication flow between the Secure API Manager components, administrative workstations, internal workstations, and external access to the API Gateway. For more information, see "Understanding the Flow of Communications through Secure API Manager" on page 18.

Table 2-3 *Secure API Manager Appliance Common Ports*

Component	Port	Description
Appliance Management	9443	Appliance management console `https://`*`lifecycle-manager-dns-name`*`:9443`
	9080	Apache/HTTPD port
	1099	Java RMI port
	80	Standard Web server ports
	25	SMTP and SMTPS outbound ports
	22	SSH port for the appliance
`ftp.novell.com`	21	Incoming port and URL required to upload the logs to the Support team. For more information, see "Sending Information to Support" in the *NetIQ Secure API Manager 1.1 Administration Guide*.

`nu.novell.com`
and
`www.novell.com`
	443	Incoming port and URLs required to register the appliance and receive product and security updates. For more information, see "Performing an Online Update" in the *NetIQ Secure API Manager 1.1 Administration Guide*.

Lifecycle Manager		

Component	Port	Description
Management console	9444	URL: `https://`*`lifecycle-manager-dns-name`*`:9444/carbon`
Administration console	9444	URL: `https://`*`lifecycle-manager-dns-name`*`:9444/admin`
Publisher	9444	URL: `https://`*`lifecycle-manager-dns-name`*`:9444/publisher`
Store	9444	URL: `https://`*`lifecycle-manager-dns-name`*`:9444/store`
Analytics		
	7613	Throttling authentication
	7713	Throttling secure transport
	9443	HTTPS servlet transport

3 Integrating Secure API Manager with Access Manager

Secure API Manager requires that you have Access Manager installed and deployed before you deploy Secure API Manager. There are IDs and tokens in Access Manager that Secure API Manager requires for you to complete the deployment.

You must create an OAuth2 application in Access Manager that allows Secure API Manager to perform OAuth2 administrative tasks on behalf of the Secure API Manager administrator. These tasks include creating, modifying, and deleting additional OAuth2 client applications (called Applications in Secure API Manager). The OAuth application also validates access tokens from the APIs. This administrative Access Manager OAuth2 client application should not be confused with the Access Manager OAuth2 client applications that you create for API grouping.

Secure API Manager is an OAuth client that retrieves the OAuth token from the OAuth application that you create in Access Manager. The Access Manager documentation contains a graphic that depicts how to implement OAuth in Access Manager. The first step of the implementation process states that you must develop a web application or REST service. The APIs in Secure API Manager are the web applications and REST services. For more information, see "Implementing OAuth in Access Manager" in the *NetIQ Access Manager 4.5 Administration Guide*.

Secure API Manager contains a Deployment Manager that walks you through deploying the different components. During the deployment of the components, you configure Secure API Manager to access your Identity Server to provide and validate the access tokens for the APIs.

There are multiple steps required to integrate Secure API Manager and Access Manager. Use the following information to create the OAuth2 application in Access Manager. You must complete these steps before deploying Secure API Manager.

- "Configuring OAuth2 in Access Manager for Use with Secure API Manager" on page 30
- "Obtaining a Long-Lived Access Token" on page 33

Configuring OAuth2 in Access Manager for Use with Secure API Manager

You must enable OAuth2 and create an OAuth2 application in Access Manager that Secure API Manager uses to obtain the OAuth tokens for the API authorizations. If you have multiple Identity Server clusters that you want Secure API Manager to reference, you must perform the following steps for each Identity Server cluster in Access Manager.

The Key Manager in the API Gateway uses this OAuth2 application to create, update, and delete OAuth2 applications and to generate tokens. This OAuth2 application must have a scope that allows full access to OAuth2 management (`urn:netiq.com:nam:scope:oauth:registration:full`) and the user associated with the token must have the roles `NAM_OAUTH2_DEVELOPER` and `NAM_OAUTH2_ADMIN` assigned.

Use the following information to enable OAuth2, create an OAuth2 application, and assign the proper rights in Access Manager.

1 Enable OAuth2 in Access Manager as follows:

- **1a** Log in to the Access Manager Administration Console.
- **1b** Click **Devices > Identity Servers >** ***IDP Cluster***.
- **1c** In the **Enabled Protocols** section, select **OAuth & OpenID Connect**.
- **1d** Click **OK**.
- **1e** Click **Update All** to update all of the Identity Servers.
- **1f** Select **All Configurations,** then click **OK** to perform the update.

2 Create a new scope for the OAuth application as follows:

- **2a** Click **Devices > Identity Servers >** ***IDP Cluster***.
- **2b** Click the **OAuth & OpenID Connect** tab.
- **2c** Click **New** to create a custom resource server for Secure API Manager.
- **2d** Specify a unique name for the resource server.
- **2e** (Conditional) If you have more than one Identity Cluster, select the appropriate Identity Cluster.
- **2f** Click **Finish**.
- **2g** Click the resource server you just created.
- **2h** Click the **Scope** tab, then click **New**.
- **2i** Use the following information to create the scope:

  Name

  Specify the name of the scope. For example, *`am_application_scope`*.

  Description

  Specify a detailed description to explain what this scope does.

  Includes claims of type

  Select **Custom Claims/Permissions** to allow Access Manager to provide the authorization tokens for the APIs in Secure API Manager.

Require user permission

Deselect this option. By not using this option, the APIs can make the calls and receive the tokens without requiring user interaction.

Allow modification in consent

Ensure that this option is not selected. By not using this option, the APIs can make the calls and receive the tokens without requiring user interaction.

2j Click **Next**.

3 Add a new, randomly named claim as follows:

3a On Step 2 of 2, click **New** to create a custom claim.

3b Specify a name for the custom claim. For example, *`APIGatewayRandomPermission`*.

3c Click **OK**.

3d Select the new claim.

3e Click **Add > Add to Access Token**.

3f Click **Finish**, then click **OK**.

4 Define the global settings as follows:

NOTE: You might have already configured the global settings for other OAuth2 applications. The following settings are the minimum settings required for Secure API Manager to work with Access Manager. For more information, see "Defining Global Settings" in the *NetIQ Access Manager 4.5 Administration Guide*.

4a On the **OAuth & OpenID Connect** tab, click the **Global Settings** tab.

4b Use the following information to define the global settings:

Authorization Grant LDAP Attribute

Specify an LDAP attribute that stores the token refresh information. This can be any attribute in the LDAP directory that accepts a long text string or use a stream attribute. For example, *`personalTitle`*.

Grant Types

Select the following options:

- **Authorization Code**
- **Implicit**
- **Resource Owner Credentials**
- **Client Credentials**

Token Types

Select the following options:

- **Access Token**
- **ID Token**
- **Refresh Token**

Token Revocation

Ensure that you deselect this option. It is enabled by default. If you revoke the Access Manager tokens, Secure API Manager cannot validate the API requests.

Access Token and ID Token Timeouts

Specify the duration in minutes for the length of time before the access token and ID token becomes invalid. Set this value to what is appropriate for your environment because this is a global setting.

Refresh Token Timeout

Specify the duration in minutes for the length of time before the refresh token becomes invalid. Set this value to what is appropriate for your environment because this is a global setting.

4c Click **Apply**.

5 Create an OAuth2 client application as follows:

5a Click **Devices > Identity Servers > Edit > OAuth & OpenID Connect > Client Applications > Register New Client**.

5b Use the following information to create the OAuth2 application:

Client Name

Specify a name for the application. For example, `Secure API Manager Administration`.

Client Type

Select **Web Based** as the client type.

Redirect URI

Specify the URI of the Access Manager Identity Server. For example:

```
https://IDP-dns=name:port/nidp/oauth2
```

Grants Required

Select all of the options except **SAML 2.0 Assertion**.

Token Types

Select all of the token types listed.

5c Click **OpenID Connect Configuration** and configure an algorithm for the Oauth token as follows:

5c1 In the **ID Token Signed Response Algorithm** field, select **RS256**.

5c2 Set the additional fields to what is appropriate for your environment. For more information, see “Defining Global Settings” in the *NetIQ Access Manager 4.5 Administration Guide*.

5d Click **Token Timeout Configuration**, then set the value of **Access Token and ID Token Timeout** to be `525600` minutes, which is one year.

5e Click **Register Client**.

5f Record the **Client ID** and **Secret** of the newly created client application so you can use them later in the Identity Server configuration in Secure API Manager.

6 Grant OAuth2 developer and administrative roles to an Access Manager administrator as follows:

6a Determine which Access Manager user is the designated OAuth2 administrator.

6b In the Access Manager Administration Console, click **Policies > Policies**.

6c Click **New** to create a new role for the OAuth2 administrator.

NOTE: You can use an existing role but you must add the following Actions to the role. For more information, see "Creating Roles" in the *NetIQ Access Manager 4.5 Administration Guide*.

6d For the Type, select Identity Server: Roles.

6e Specify a detailed description for the policy so it is easy to remember that it is the policy for Secure API Manager access.

6f In the Condition Group, click the New drop-down menu, then select LDAP Attribute.

6g In the LDAP Attribute field, click GUID, then find and select cn.

6h In the Value field, click LDAP Attribute, then find and select Data Entry Field.

6i Specify the name of the administrator user that is the administrator for OAuth in your Access Manager environment.

6j In the Actions section, select Activate Role, then add the following two roles:

- `NAM_OAUTH2_DEVELOPER`
- `NAM_OAUTH2_ADMIN`

6k Click OK twice.

6l Click Apply Changes, then click Close.

6m (Conditional) If you created a new policy, click Edit IDP > Roles > *Select the new policy* > Enable. If the policy does not appear in the list, click Manage Policies, then click the new policy to enable it.

6n Click Save to create the new policy or enable an existing policy.

7 Update all Identity Servers with the configuration changes as follows:

7a In the Access Manager Administration Console, click Identity Servers.

7b Click Update All to reconfigure all of the nodes in the cluster for the Identity Servers.

Obtaining a Long-Lived Access Token

After you have created an OAuth2 application and configured Access Manager for OAuth2, you must generate a long-lived access token that Secure API Manager uses to authenticate the APIs. You must obtain this long-lived access token from each Identity Server cluster.

1 Log in to the appliance management console as `vaadmin` on any appliance.

`https://`*`appliance-dns-name`*`:9443`

2 Click Deployment Manager to access the Deployment Manager.

3 Click the Access Manager Integration tab.

4 In the Access Manager Identity Server DNS Name and Access Manager Identity Server Port fields, provide the DNS name and port for your Access Manager Identity Server.

5 Click Auto-Fill to have the Deployment Manager populate the fields with the information from Access Manager.

6 In the Client Id field and the Client Secret field, enter the Client ID and Secret that you recorded in Step 5f on page 32.

7 Click **Get Token** to generate the long-live access token.

8 Click **Save** to save the configuration information.

You can now deploy Secure API Manager using the information from the OAuth2 application. You complete the integration of Secure API Manager and Access Manager during the deployment of Secure API Manager.

4 Deploying Secure API Manager

This section guides you through the process of deploying the Secure API Manager appliances that become different components of Secure API Manager. You download a single Secure API Manager appliance that has the ability to become any combination of the four Secure API Manager components. Ensure that you have chosen the best deployment scenario for your environment before continuing. For more information, see "Understanding Deployment Scenarios" on page 19.

To properly deploy Secure API Manager requires several separate processes.

- **Creating the OAuth2 application in Access Manager:** You must have created an OAuth2 application in Access Manager before you can deploy Secure API Manager. For more information, see Chapter 3, "Integrating Secure API Manager with Access Manager," on page 29.
- **Deploying the appliances:** You must deploy the appropriate number of appliances in VMware that you need to use in your Secure API Manager configuration. When you deploy the appliances, you configure the time and networking settings for each appliance. The appliances must be up and running in VMware with IP addresses and DNS names assigned to them before you can deploy the Secure API Manager components using the Deployment Manager. For more information, see "Deploying a Secure API Manager Appliance" on page 36.
- **Collecting the required information:** To properly deploy and configure the different Secure API Manager components using the Deployment Manager, you must have gathered the required information before starting the Deployment Manager. You must have the DNS name of each appliance, the NFS server information, load balancer information for each cluster, and the Access Manager configuration information. Fill out the deployment worksheet before running the Deployment Manager. For more information, see "Secure API Manager Deployment Worksheet" on page 40.
- **Using the Deployment Manager:** After you have deployed the appliances in VMware and collected the required information, you can now run the Deployment Manager to create and configure the different Secure API Manager components. The Deployment Manager requires that you install the components in a certain order. For more information, see "Understanding the Secure API Manager Deployment Manager" on page 38.

Deploying the Secure API Manager Appliances

VMware is the only supported platform for Secure API Manager. We recommend that you have a good understanding of VMware before deploying the appliance. This guide does not contain instructions for using VMware or how to deploy appliances in VMware. Currently, the appliance is not supported in Amazon Web Service or Azure environments. For more information, see the VMware Docs (https://docs.vmware.com/) website.

Each appliance has its own administrative user of `root`. You set the password for the `root` user when you deploy each appliance. It is important to have a record of the IP address, DNS name, and login information for each appliance. You can enable an additional administrative account after you deploy the appliance. For more information, see "Setting Administrative Passwords" in the *NetIQ Secure API Manager 1.1 Administration Guide*.

Use the following sections to deploy the appliances and record the appliance information for your environment.

- "Deploying a Secure API Manager Appliance" on page 36
- "Recording the IP Addresses, DNS Names, and Login Information for the Appliances" on page 37

Deploying a Secure API Manager Appliance

You must deploy one or more appliances that will contain one or more Secure API Manager components. When you deploy the appliance, you set the time zone of the appliance, configure the network settings for the appliance, and create a password for the `root` user of the appliance.

Secure API Manager uses Docker containers to create the different components. After you define the appliance-specific setting, the initialization process extracts the Docker containers for each component on the appliance.

IMPORTANT: The extraction process can take 30 minutes or longer to complete. Ensure that you wait for the appliance to complete the extraction process before configuring each component.

Each DNS name of the appliance must be publicly resolvable, even in test environments. The DNS name of each appliance must be publicly resolvable to allow the Docker containers access to the local `/etc/hosts` file of the appliance. If the DNS name is not publicly resolvable, the components cannot communicate with each other and the product does not work.

To deploy a Secure API Manager appliance:

1. Ensure that you have determined the number of appliances you will need for your environment. For more information, see "Understanding Deployment Scenarios" on page 19.
2. Download the appliance file from the Customer Center. For more information, see "Obtaining Secure API Manager" on page 25.
3. Deploy the appliance to your virtual environment. For more information, see Deploy an OVF or OVA Template. (https://docs.vmware.com/en/VMware-vSphere/6.5/com.vmware.vsphere.vm_admin.doc/GUID-17BEDA21-43F6-41F4-8FB2-E01D275FE9B4.html).
4. Power on the appliance.
5. Select the appropriate language, then read the license and click **Accept**.
6. Use the following information to configure basic settings for the appliance:

root Password

Specify a password for the `root` user on the appliance.

NTP Server

Specify a primary and secondary NTP server used to keep time on the appliance.

Region and Time Zone

Select your region and time zone.

Hostname and Networking options

Specify a host name for the appliance, then select whether to use a static IP address or DHCP. If you use a static IP address, you must specify the IP address, subnet mask, the gateway, and the DNS servers.

7 Click **Finish** and wait for the appliance initialization to complete.

IMPORTANT: The initialization process can take 30 minutes or longer to complete. The initialization process extracts the images of the components.

8 Record the IP address, DNS name, and login information for future reference and for use during the deployment of the Secure API Manager components. For more information, see "Recording the IP Addresses, DNS Names, and Login Information for the Appliances" on page 37.

9 Repeat Step 3 through Step 8 for each appliance you must deploy.

After you have the appropriate number of appliances for your Secure API Manager environment, you must deploy the appropriate components on one or more of the appliances using the Deployment Manager. You must understand the Deployment Manager before trying to use it. For more information, see "Understanding the Secure API Manager Deployment Manager" on page 38.

Recording the IP Addresses, DNS Names, and Login Information for the Appliances

Each Secure API Manager appliance uses SUSE Linux Enterprise Server as the operating system. During the deployment of the appliance, you set the password for the **root** user and define your networking settings for the appliance.

It is very important that you keep a record of the IP address, DNS name, and login information for each appliance you deploy. You configure and manage each appliance through the appliance management console. The login for the appliance management console is the IP address or DNS name of the appliance at port 9443. You log in using the `root` user and the password you specify during the deployment of the appliance. Each appliance has its own password.

WARNING: There is no way to reset or retrieve the `root` password. If you forget or lose the `root` password, your only option is to delete the appliance from the virtual environment and redeploy a new appliance.

Ensure that you have the correct network settings assigned to the appliances. If you deploy a component and want to change the network settings later, Secure API Manager does not see the changes to the network settings. The IP addresses and DNS names are stored in the Database Service component and on the file system stored on the NFS server.

If you must change the network settings on an appliance at a later time, you must delete the component from Secure API Manager, delete the appliance, then redeploy the appliance with the correct network settings.

Use the following worksheet to record your appliance login information.

Table 4-1 *Worksheet for Appliance Login Information*

Component	IP Address:Port	DNS Name:Port	Login Information
API Gateway			
API Gateway cluster member			
API Gateway cluster member			
Analytics			
Analytics cluster member			
Analytics cluster member			
Database Service			
Database Service cluster member			
Database Service cluster member			
Lifecycle Manager			
Lifecycle Manager cluster member			
Lifecycle Manager cluster member			

The extra lines in the worksheet are for clustering the different components. For more information, see "Enabling High Availability and Load Balancing" on page 23.

Understanding the Secure API Manager Deployment Manager

Secure API Manager provides a Deployment Manager that walks you through deploying all of the components in a single process to make the deployment process simpler. This allows you to configure the appropriate number of appliances and clusters for your environment at the same time.

The Deployment Manager does more than deploy Secure API Manager. It also validates the communication between appliances, integrates Access Manager with Secure API Manager, and provides an overview of all of the appliances in your Secure API Manager deployment.

- "Validating Communication Between Appliances" on page 39
- "Understanding the Deployment Options" on page 39
- "Viewing the Secure API Manager System" on page 40

Validating Communication Between Appliances

All of the communication for the Deployment Manager takes place over SSL. Secure API Manager requires secure communication to ensure that no security issues occur when the components communicate or when you use the Deployment Manager. The Deployment Manager requires:

- A trusted root certificate or a self-signed certificate for the SSL communication.
- DNS names and IP addresses assigned to the appliances for your Secure API Manager system. For more information, see "Deploying the Secure API Manager Appliances" on page 36.
- All components must have direct access to the primary database without going through an L4 switch or database corruption can occur.

WARNING: Do not edit the appliance configuration settings during the deployment of the Secure API Manager components. The Deployment Manager stores information on each appliance. If you edit the appliance configuration settings through the appliance management console while the Deployment Manager is configuring other components, you corrupt the components and the deployment fails. Wait until the Deployment Manager has finished configuring all of the components before editing the appliance configuration settings.

Understanding the Deployment Options

The Deployment Manager allows you to create a new system or join an existing system. The Deployment Manager also expects you to deploy and configure all of the appliances and components in your system through one process. The options you select during the deployment.

If you are creating a new system, you must always deploy the Database Service component first. The Database Service component stores the configuration information for the other components to ensure that the data and the configuration information is correct on each node depending on your specific environment and goals.

When you deploy each component or each node of a cluster, Secure API Manager stores that information in the Database Service component. To ensure that the correct information is available and synchronized properly, at the end of a deployment of an appliance you have three different options on how to proceed.

The Deployment Manager provides the following options to allow you to choose what happens to the entire system:

- **Save configuration:** Select this option if you are deploying multiple appliances at the same time. This option saves the configuration information for the appliance in the database but it does not actually deploy the components and no services are available on this appliance at this time.

- **Save configuration and deploy only this appliance:** Select this option if this is the first Database Service component you are deploying or if you must replace a single node in a configured Secure API Manager system. This option saves the configuration file to the database and deploys this appliance.

  If you are replacing a failed node, this option might leave other appliances in an invalid state because they might need to be reconfigured to know about this new appliance. For more information, see "Recovering from a Failed Node in a Cluster" on page 64.
- **Save configuration and reconfigure the entire system:** Select this option to save the configuration in the database, deploy the appliance, and redeploy all other appliances in the system to receive the updated configuration information. You would use this option when you are deploying the last appliance in the system and you need to reconfigure every other appliance in the system to know about all of the other appliances.

Ensure that you collect all of the information listed in Table 4-2 on page 41 before you use the Deployment Manager to deploy Secure API Manager.

Viewing the Secure API Manager System

The Deployment Manager also provides a **STATUS** tab that displays what happens when you deploy a component or deploy a component and reconfigure the entire system. Depending on which option you select at the end of the configuration of a new component, the Deployment Manager automatically takes you to the **STATUS** tab.

The Deployment Manager also contains an **SYSTEM** tab. The **SYSTEM** tab allows you to view all of the appliances in the Secure API Manager system from one page. The information on the **SYSTEM** tab is global. It does not matter which appliance management console you access, you see the same information for your system.

The **SYSTEM** tab also displays which node is the primary Database Service component. There must be a primary Database Service component to store the configuration information. If the primary Database Service component fails, you can make another Database Service component the primary on the **SYSTEM** tab. You can have only one primary Database Service component at a time.

Secure API Manager Deployment Worksheet

Use the following worksheet to gather the information you must have to complete the Secure API Manager deployment. The required information is different depending on your deployment scenario. The **Value** column provides a place for you to record the information before you start the deployment.

There are a few important items to understand before you deploy Secure API Manager:

- You must have deployed the number of Secure API Manager appliances that you will use in your deployment before you launch the Deployment Manager. For more information, see "Deploying the Secure API Manager Appliances" on page 36.
- If you do not have port forwarding to port 9444-enabled components on the load balancers, you must always specify the DNS name of the primary Database Service node.

- Ensure that the appliances have a publicly resolvable DNS name. Otherwise, deployment fails, even for test environments.
- Ensure that the components have direct access to the primary database without going through an L4 switch or database corruption can occur.

Table 4-2 Secure API Manager Worksheet

Item	Value	Notes
Trusted root certificate or self-signed certificate		You must have a trusted root certificate or a self-signed certificate for the Deployment Manager to work properly. The Secure API Manager components communicate over SSL. For more information, see "Understanding the Secure API Manager Deployment Manager" on page 38.
Shared Storage NFS Server		(Recommended) For production environments, we recommend that you cluster the Secure API Manager components you deploy. If you cluster the components, you must have an NFS server deployed and running in your IT environment before running the Deployment Manager. For more information, see "Using High Availability and Load Balancing with Secure API Manager" on page 22.
Shared Storage > NFS Host IP Address		The IP address of your NFS server in your IT environment. You must have an NFS server to maintain the configuration information for Secure API Manager when you cluster the nodes. For more information, see "Configuring Content Synchronization for High Availability Using a Network File System Server" on page 24.
Shared Storage > NFS Mount		The path to the mount point on the NFS server where Secure API Manager stores the configuration information in a clustered environment. For more information, "Using High Availability and Load Balancing with Secure API Manager" on page 22.

Item	Value	Notes
Logging (Syslog)		(Optional) Add the Syslog information if you need Secure API Manager to send the log information to a Syslog server you have installed in your IT environment. These settings are global. When you set the value on one component, all of the other components or nodes in a cluster receive this information.
Logging > Remote Syslog Host		The DNS name of the Syslog server that you want to use to capture the information from Secure API Manager.
Logging > Remote Syslog Port		The port that the Syslog server uses to communicate with Secure API Manager.
Logging > Protocol		The protocol that the Syslog server uses. It is either TCP or UDP.
Database Service Component		You must deploy the Database Service component first. The Database Service deploys the proper database for Secure API Manager to store configuration, user, and API information. **WARNING:** The Database Service component must run on its own appliance. Do not combine any other components with the Database Service component.
Database Service Component > Load Balancer Host		(Recommended) The DNS name of the load balancer host for the Database Service cluster. You must have a load balancer for each component you cluster.
Database Service Component > Database User		A user name for the database administrative user for the Database Service. You use this account to join other appliances to your Secure API Manager system.
Database Service Component > Database Password		A password for the database administrative user.

Item	Value	Notes
Database Service Component > Database Host		The DNS name of the appliance that you need to become the primary Database Service component.
Database Service Component > Database Host		(Recommended) The DNS name of the appliance that you need to become a node in the Database Service cluster. For a cluster, you need two or three nodes.
Database Service Component > Database Host		(Recommended) The DNS name of the appliance that you need to become a node in the Database Service cluster. For a cluster, you need two or three nodes.
Lifecycle Manager		Contains the different consoles for Secure API Manager.
Lifecycle Manager > Load Balancer Host		(Recommended) The DNS name of the load balancer host for the Lifecycle Manager cluster. You must have a load balancer for each component you cluster.
Lifecycle Manager > Lifecycle Manager Host		The DNS name of the appliance that you need to become the Lifecycle Manager component.
Lifecycle Manager > Lifecycle Manager Host		(Recommended) The DNS name of the appliance that you need to become a node in the Lifecycle Manager cluster. For a cluster, you need two or three nodes.
Lifecycle Manager > Lifecycle Manager Host		(Recommended) The DNS name of the appliance that you need to become a node in the Lifecycle Manager cluster. For a cluster, you need two or three nodes.
Gateway		Directs traffic to and from the APIs.
Gateway > Load Balancer Host		(Recommended) The DNS name of the load balancer host for the API Gateway cluster. You must have a load balancer for each component you cluster.
Gateway > Gateway Host		The DNS name of the appliance that you need to become the API Gateway component.

Item	Value	Notes
Gateway > Gateway Host		(Recommended) The DNS name of the appliance that you need to become a node in the API Gateway cluster. For a cluster, you need two or three nodes.
Gateway > Gateway Host		(Recommended) The DNS name of the appliance that you need to become a node in the API Gateway cluster. For a cluster, you need two or three nodes.
Analytics		Provides reports about API usage and statistics. We recommend that Analytics always runs on its own appliance.
Analytics > Load Balancer Host		(Recommended) The DNS name of the load balancer host for the Analytics cluster. You must have a load balancer for each component you cluster.
Analytics > Analytics Host		The DNS name of the appliance that you need to become the Analytics component.
Analytics > Analytics Host		(Recommended) The DNS name of the appliance that you need to become a node in the Analytics cluster. For a cluster, you need two or three nodes.
Analytics > Analytics Host		(Recommended) The DNS name of the appliance that you need to become a node in the Analytics cluster. For a cluster, you need two or three nodes.
Access Manager Integration		You must perform the integration with Access Manager for Secure API Manager to work. Ensure that you have created the OAuth2 application in Access Manager. For more information, see Chapter 3, "Integrating Secure API Manager with Access Manager," on page 29.

Item	Value	Notes
Trusted Root Certificate from Access Manager		You must import the trusted root certificate that you used to secure Access Manager into Secure API Manager to complete the Access Manager configuration. For more information, see "Managing Trusted Roots and Trust Stores" in the *NetIQ Access Manager 4.5 Administration Guide.*
Name		A display name for the Access Manager Identity Server that appears in the Deployment Manager.
Description		A description of the Access Manager Identity Server. This allows you to provide additional information about the Identity Server so that other people will know which Access Manager Identity Server this is.
Discovery Endpoint		The Deployment Manager displays the format for the Access Manager discovery endpoint. If you populate this field correctly and import the certificate during the deployment, the Deployment Manager can auto-populate the remaining fields with the correct information.

Deploying a Test System

Secure API Manager allows you to deploy all four components on one appliance only for testing purposes. You cannot cluster a test system. The steps to configure a test system are different from those of a production system. You must deploy one virtual appliance before you can deploy the test system. For more information, see "Deploying the Secure API Manager Appliances" on page 36.

1 Access the appliance management console using the `root` user and password you set when you deployed the appliance. For more information, see "Deploying the Secure API Manager Appliances" on page 36.

   ```
   https://ip-address-or-dns-name-appliance:9443
   ```

2 Click **Deployment Manager**.

3 Click **Create**.

4 Click **Test System**.

5 Copy the appliance host name that the Deployment Manager displays in the upper left corner of the screen.

6 Use the following information to configure the Database Service:

Load Balancer Host

Paste the appliance host name in this field.

Database User

Specify a name for the database administrator of the Database Service.

Password

Specify a password for the database administrator.

Database Host

Paste the appliance host name in this field.

7 Click **Next**.

8 Use the following information to configure the Lifecycle Manager:

Load Balancer Host

Paste the appliance host name in this field.

NFS options

Do not specify any information in these fields. These fields are only for clustered deployments. Specifying information in these fields causes the deployment to fail.

Lifecycle Manager Host

Paste the appliance host name in this field.

9 Click **Next**.

10 Use the following information to configure the API Gateway:

Load Balancer Host

Paste the appliance host name in this field.

NFS Options

Do not specify any information in these fields. These fields are only for clustered deployments. Specifying information in these fields causes the deployment to fail.

Gateway Host

Paste the appliance host name in this field.

11 Click **Next**.

12 Use the following information to configure Analytics:

Load Balancer Host

Paste the appliance host name in this field.

Analytics Host

Paste the appliance host name in this field.

13 Click **Save and deploy**.

14 Click the **STATUS** tab to watch the deployment. It can take up to 15 minutes for the Deployment Manager to deploy and configure the test system.

15 Wait until you see a green check mark in the **Success** column.

16 Click the **SYSTEM** tab and ensure that there are green check marks beside the components.

The test system is up and running and now you must finish the Access Manager integration. For more information, see "Completing the Integration Between Secure API Manager and Access Manager" on page 51.

Deploying the Secure API Manager Components

Secure API Manager provides a Deployment Manager that walks you through deploying all of the components. The Deployment Manager resides in the appliance management console. You must have deployed the appropriate number of virtual appliances for your configuration before using the Deployment Manager. For more information, see "Deploying the Secure API Manager Appliances" on page 36.

Use the information you collected in Table 4-1, "Worksheet for Appliance Login Information," on page 38 to deploy the Secure API Manager components. The Deployment Manager deploys all of the components during this process. You must always deploy the Database Service component first. It stores all of the configuration information for the entire system.

The following procedure assumes that you are deploying each component on a separate appliance and that you are clustering each component.

To deploy a production system:

1 Ensure that you have the correct network settings for the appliances. If you have to change the network settings later, you must delete the component from the Secure API Manager system, delete the appliance, redeploy the appliance, then redeploy the component. For more information, see "Recording the IP Addresses, DNS Names, and Login Information for the Appliances" on page 37.

2 Ensure that all components have direct access to the primary database without going through an L4 switch or database corrupts can occur.

3 Ensure that the load balancers use sticky sessions. Otherwise, the load balancers allow the different components to corrupt the information in the Database Service component. For more information about load balancing, see "Using High Availability and Load Balancing with Secure API Manager" on page 22.

4 Access the appliance management console for the appliance that you need to become the first Database Service component. Use the `root` user and password that you set during the deployment of the appliance. For more information, see "Deploying the Secure API Manager Appliances" on page 36.

```
https://ip-address-or-dns-name-appliance:9443
```

5 Click **Deployment Manager**.

6 Click **Create**.

7 Click **Database**.

8 Create the primary Database Service component.

8a Specify the information for the Database Service component using the information you gathered in the worksheet. For more information, see Chapter 4, "Deploying Secure API Manager," on page 35.

IMPORTANT: Remember the user name and password you define for the database administrative user. You use this account to add the additional components to the Secure API Manager system. In addition, you use this account to access the database through an SQL client, when needed.

8b Click **Save Configuration And Deploy**.

8c Watch the status of the deployment of the Database Service component on the **STATUS** tab. The Deployment Manager automatically takes you to the **STATUS** tab.

8d When the **STATUS** tab states that the deployment is complete, click the **SYSTEM** tab to ensure that there is a star next to it to designate that this is the primary node in the cluster.

9 Deploy the second Database Service component.

9a Log in to the appliance management console on a second appliance that you need to become the second node of the Database Service component cluster. Use the root account and password you set for this second appliance.

`https://`*`ip-address-or-dns-name-appliance`*`:9443`

9b Click **Deployment Manager**.

9c Click **JOIN EXISTING** to add this node to the new deployment.

9d Approve the certificate that the Deployment Manager displays or import a trusted root certificate for this appliance.

9e Specify the DNS name for the first Database Service node and specify the database user name and password you created in Step 8a.

IMPORTANT: If you do not have port forwarding enabled to port 9444 on the load balancers, you must always specify the DNS name of the primary Database Service node.

9f Click **JOIN** and wait for this node to join the existing node.

9g Click **GO TO CONFIGURATION** and add the configuration information for this node.

9h In the **Database Host** field, specify the DNS name or IP address of this appliance.

9i Click **Next** three times.

9j Click **SAVE**.

9k Select **Save configuration only**, then click **Save**.

This saves the configuration file but does not deploy the component at this time. This option reduces the number of times an appliance has to be restarted during the deployment of the entire system.

9l (Conditional) If you want to deploy a third Database Service node, repeat Step 9a through Step 9k for this last node.

10 (Optional) Configure Logging.

You can configure logging at any time during the deployment when you are on the Database configuration page. These options are global and you have to perform them on only one appliance.

10a In the Deployment Manager, click **Logging** on the Database configuration page.

10b Select **Enable**.

10c Specify the IP address or DNS name of your Syslog server, the port, and the protocol it uses.

11 Deploy the Lifecycle Manager component.

11a Access the appliance management console for the appliance that you need to become the Lifecycle Manager component. Use the `root` user and password that you set during the deployment of the appliance. For more information, see "Deploying the Secure API Manager Appliances" on page 36.

```
https://ip-address-or-dns-name-appliance:9443
```

11b Click **Deployment Manager**.

11c Click **Join**.

11d Specify the DNS name for the primary Database Service node and specify the database user name and password you created in Step 8a.

IMPORTANT: If you do not have port forwarding to port 9444 enabled on the load balancers, you must always specify the DNS name of the primary Database Service node.

11e Click **Join** and wait for this node to join the system.

11f Approve the certificate that the Deployment Manager displays or import a trusted root certificate for this appliance.

11g Do not specify any information on the Database Deployment page, then click **Next**.

11h On the Lifecycle Manager Deployment page, use the information you gathered in the worksheet for the Lifecycle Manager, such as the NFS server information, to configure the Lifecycle Manager component. For more information, see Table 4-1 on page 38.

11i Click **Next** twice.

11j Click **Save**.

11k Select **Save configuration only**, then click **Save**.

This saves the configuration file but does not deploy the component at this time. This option reduces the number of times the Deployment Manager restarts an appliance during the deployment of the entire system.

12 Repeat Step 11 for each additional Lifecycle Manager node that you need to deploy.

13 Deploy the API Gateway component.

13a Access the appliance management console for the appliance that you need to become the API Gateway component. Use the `root` user and password that you set during the deployment of the appliance. For more information, see "Deploying the Secure API Manager Appliances" on page 36.

```
https://ip-address-or-dns-name-appliance:9443
```

13b Click **Deployment Manager**.

13c Click **Join**.

13d Approve the certificate that the Deployment Manager displays or import a trusted root certificate for this appliance.

13e Specify the DNS name for the primary Database Service node and specify the database user name and password you created in Step 8a.

IMPORTANT: If you do not have port forwarding to port 9444 enabled on the load balancers, you must always specify the DNS name of the primary Database Service node.

13f Click **Join** and wait for this node to join the system.

13g Do not specify any information on the Database Deployment page, then click **Next**.

13h Do not specify any information on the Lifecycle Manager Deployment page, then click **Next**.

13i On the Gateway Deployment page, use the information you gathered in the worksheet to configure the API Gateway component. For more information, see Table 4-2 on page 41.

13j Click **Next**.

13k Click **Save**.

13l Select **Save configuration only**, then click **Save**.

This saves the configuration file but does not deploy the component at this time. This option reduces the number of times the Deployment Manager restarts an appliance during the deployment of the entire system.

14 Repeat Step 13 for each additional API Gateway node that you need to deploy.

15 Deploy the Analytics component.

15a Access the appliance management console for the appliance that you need to become the Analytics component using the `root` user and password you set during the deployment of the appliance. For more information, see "Deploying the Secure API Manager Appliances" on page 36.

```
https://ip-address-or-dns-name-appliance:9443
```

15b Click **Deployment Manager**.

15c Click **Join**.

15d Approve the certificate that the Deployment Manager displays or import a trusted root certificate for this appliance.

15e Specify the DNS name for the primary Database Service node and specify the database user name and password you created in Step 8a.

IMPORTANT: If you do not have port forwarding to port 9444 enabled on the load balancers, you must always specify the DNS name of the primary Database Service node.

15f Click **Join** and wait for this node to join the system.

15g Do not specify any information on the Database Deployment page, then click **Next**.

15h Do not specify any information on the Lifecycle Manager Deployment page, then click **Next**.

15i Do not specify any information on the Gateway Deployment page, then click **Next**.

15j On the Analytics Deployment page, use the information that you gathered in the worksheet to configure the Analytics component. For more information, see Table 4-1 on page 38.

15k Click **Save**.

15l Select **Save configuration only**, then click **Save**.

This saves the configuration file but does not deploy the component at this time. This option reduces the number of times an appliance has to be restarted during the deployment of the entire system.

16 Repeat Step 15 for each additional Analytics node you need to deploy except for the last node.

17 On the last Analytics node, click **Save configuration, deploy this appliance, and reconfigure the entire system**.

18 On the **STATUS** tab of the primary Database Service node, watch the deployment and reconfiguration of each appliance in the system. This process can take time depending on the number of nodes you deployed.

After the deployment finishes, you must complete the integration with Access Manager to complete the deployment and have a fully functioning system. For more information, see “Completing the Integration Between Secure API Manager and Access Manager” on page 51.

Completing the Integration Between Secure API Manager and Access Manager

To finish the deployment of Secure API Manager you must complete the integration with Access Manager. Ensure that you have created the OAuth2 application in Access Manager before proceeding. For more information, see Chapter 3, “Integrating Secure API Manager with Access Manager,” on page 29.

The remaining task is to configure Secure API Manager to access and use the Access Manager OAuth2 application. You perform these steps on only one appliance to complete the integration. The configuration steps you perform and information on the Access Manager Integration tab are global. It does not matter which appliance management console you use to complete the following task.

NOTE: Access Manager uses the term **Identity Server** and Secure API Manager uses the term **Identity Provider**. However, both terms refer to the Access Manager Identity Server.

To configure Secure API Manager to use the Access Manager OAuth2 application:

1 Log in to the appliance management console for any appliance using the `root` user and password you set during the deployment of the appliance. For more information, see “Deploying the Secure API Manager Appliances” on page 36.

```
https://ip-address-or-dns-name-appliance:9443
```

2 Click **Deployment Manager** to launch the Deployment Manager.

3 Click the **Access Manager Integration** tab.

4 Use the following information to define your Access Manager Identity Server.

Name

Specify a display name for the Access Manager Identity Server that appears in the Deployment Manager.

Description

Specify a description of the Access Manager Identity Server. This allows you to provide additional information about the Identity Server so that other people will know which Access Manager Identity Server this is.

Identity Server DNS Name

Specify the DNS name of your Access Manager Identity Server.

Identity Server Port

Specify the port for your Access Manager Identity Server.

Endpoints

Click **Auto-Fill** to accept the Access Manager certificate and complete the endpoint fields automatically.

Client ID

Specify the **Client ID** that you recorded in Step 5f on page 32.

Client Secret

Specify the **Secret** that you recorded in Step 5f on page 32.

Access Token

Specify the long-lived access token you generated in "Obtaining a Long-Lived Access Token" on page 33.

5 Click **Save** to save the configuration information and register the Access Manager Identity Server with Secure API Manager.

Post-Deployment Steps

After you have deployed Secure API Manager on one or more appliances, you must perform some post-deployment steps. Some of the steps are appliance-specific and some of the steps are for Secure API Manager.

1. Record the IP address, DNS name, and login information for each appliance. When you deployed the appliance, you set these values. Creating a record helps you in the future when you have to apply patches or perform any additional administrative work for the appliance. For more information, see "Recording the IP Addresses, DNS Names, and Login Information for the Appliances" on page 37.
2. Secure API Manager contains only one administrative global user account. Secure API Manager is an appliance and every deployment of Secure API Manager contains this same user with the same password. The user is `admin` and the password is `admin`. You must log in to the appliance management console for Secure API Manager and change the password. For more information, see "Changing the Default Password for the Administrator" in the *NetIQ Secure API Manager 1.1 Administration Guide*.
3. By default, the only user you can use to access the appliance management console is `root`. You set the password for the `root` user when you deployed the appliance. There is an additional administrative user for the appliance with the name of `vaadmin`. You must set a password for the `vaadmin` user before you can use it to log in to the appliance management console. Log in to the appliance management console as `root`, then set a password for the `vaadmin` user. You must do this for each appliance you deploy. For more information, see "Setting Administrative Passwords" in the *NetIQ Secure API Manager 1.1 Administration Guide*.
4. Log in to each appliance management console and register the appliance to receive security and product updates. For more information, see "Performing an Online Update" in the *NetIQ Secure API Manager 1.1 Administration Guide*.
5. Create user accounts for the API developers to access the Publisher and Store so they can create APIs in a single location. For more information, see "Managing Users" in the *NetIQ Secure API Manager 1.1 Administration Guide*.

5 Troubleshooting Your Deployment

Secure API Manager provides several tools to help you troubleshoot issues with your deployment. You can turn logging on or off as needed for certain components, configure various log management settings to manage disk space, and download log files to send to NetIQ Technical Support. You can also reset an appliance or an entire system if you want to completely restart the deployment process without having to deploy new VMs.

Configuring and Managing Deployment Logging

Secure API Manager allows you to configure various settings for the deployment log files for each product component. Since log files can quickly grow and take up valuable disk space, Secure API Manager bundles and prunes them according to the settings you configure. Bundling log files compresses them to reduce their size, helping you conserve disk space on the appliance.

1 Log in to the appliance management console as the `vaadmin` user.

   `https://`*`ip-address-or-dns-name-appliance`*`:9443`

2 Click **Deployment Manager**, then select **Settings** from the menu in the upper right corner of the page.

3 Under **Logging**, turn logging on or off as needed for the Deployment Manager, Lifecycle Manager, or Gateway.

4 (Conditional) If you want to download the Deployment Manager log files to send to NetIQ Technical Support, click the down arrow next to **Deployment Manager Logging**.

5 Under **Logging Disk Space**, review the log bundle settings and customize them as needed for your environment.

   NOTE: Any changes you make to the following settings do not take effect until the Bundle Creation Period is over or you reboot the system. For example, with the default Bundle Creation Period of 5 hours, your changes would not take effect until the end of that 5 hour period.

   Bundling Creation Period The frequency with which Secure API Manager checks whether any of the maximum values that you configured on this page have been exceeded. The default is 5 hours. If your system is functioning well, your log files should not grow too quickly, so you might be able to increase the polling period.

   Maximum <*Component*> Log Size The maximum size that each component log can reach before Secure API Manager begins bundling it.

Maximum Bundle Size When the overall log bundle size for all components exceeds this value, Secure API Manager runs a pruning process, beginning with the oldest logs.

Maximum Bundle Count When the total number of log bundles for all components exceeds this value, Secure API Manager runs a pruning process, beginning with the oldest logs.

6. Click **Save**.

The **Log Bundles** section shows the directory location of the log bundles and also displays a table containing the latest details about the log bundles that have been collected for each component. The table is empty until the component logs are large enough to be bundled. Next to the **Log Bundles** heading the following options are available:

- To update the information in the table, click the **Refresh** icon.
- To immediately create log bundles, click the **Bundle Log Files Now** icon. This action uses only the **Maximum Bundle Size** and **Maximum Bundle Count** settings.
- To immediately delete all the log bundles, click the **Delete Log Bundles** icon.

Configuring System Messages

Secure API Manager provides system messages that are generated during deployment and when certain events occur. NetIQ Technical Support can use these messages to diagnose issues. The configuration options allow you to specify how many of each type of message to keep and which messages to keep.

1. Log in to the appliance management console as the `vaadmin` user.

   `https://`*`ip-address-or-dns-name-appliance`*`:9443`

2. Click **Deployment Manager**, then select **Advanced** from the menu.
3. Next to the **System Messages** heading, click the **Settings** icon.
4. Specify the appropriate message settings for your environment, then click **Save**.

Viewing Gateway Statistics

Secure API Manager provides details of gateway connections and latency in your environment.

NOTE: The Gateway Connections and Gateway Latency pages display data only if you are logged on to an appliance that has a gateway deployed.

1. Log in to the appliance management console as the `vaadmin` user.

   `https://`*`ip-address-or-dns-name-appliance`*`:9443`

2. Click **Deployment Manager**, then select **Advanced** from the menu.
3. (Optional) Click the **Gateway Connections** tab to view data about HTTP and HTTPS connections on the gateway.
4. (Optional) Click the **Gateway Latency** tab to view latency information.

Resetting Appliances

The Deployment Manager provides the ability to reset one or more appliances, or even your entire system. For example, you might need to reset one or more appliances if the components are not communicating as they should. Resetting an appliance means that you completely remove the selected component and restart the deployment process without having to redeploy the OVF file on the VM and set up networking again. This option essentially takes you back to the point where you have not yet run the Deployment Manager.

WARNING: Whether you reset a single appliance or all appliances in the system, this action removes all component settings, data, and configuration. You cannot undo this action.

1. Log in to the appliance management console as the `vaadmin` user.

   `https://`*`ip-address-or-dns-name-appliance`*`:9443`

2. Click **Deployment Manager**, then select **Advanced** from the menu.
3. Under **Reset**, choose one of the following options:

   Reset Appliance Resets the appliance where you are currently logged in.

   Reset All Appliances Resets the appliance of every node in your system to its default.

4. Read the warning message, then click **Proceed** to continue.

The Deployment Status page shows the status of the appliance reset operation.

Checking NFS or Syslog Server Status

The Deployment Manager allows you to check configured NFS and Syslog servers to help determine if there any issues with their connections.

1. Log in to the appliance management console as the `vaadmin` user.

   `https://`*`ip-address-or-dns-name-appliance`*`:9443`

2. Click **Deployment Manager**, then click **Status** on the top bar.
3. From the **Check Connection** menu in the top right corner of the screen, select **NFS** or **Syslog**.

The Deployment Status page displays the status of the NFS and Syslog server tests.

6 Uninstalling Secure API Manager

To uninstall Secure API Manager, power off the appliance for each component and then delete the image from your virtual environment. If you have clustered the components using an L4 switch, ensure that you remove the IP addresses of the components from the L4 switch.

This deletes Secure API Manager from your IT environment. You can now redeploy the Secure API Manager system. For more information, see “Deploying the Secure API Manager Components” on page 47.

7 Preparing Secure API Manager for a Disaster

Secure API Manager stores all configuration information for the system in the Database Service component. It is important that you cluster the Database Service component to ensure high availability. If one node in a cluster goes down, it is a simple process to redeploy that node as long as there is one Database Service node up and running. For more information, see "Using High Availability and Load Balancing with Secure API Manager" on page 22.

It is also very important that you create and implement a backup plan for the Database Service component. If a disaster occurs to your entire system, you can restore the Database Service component and then redeploy the other components to restore your system.

Preparing the Database Service Component for a Disaster

It is very important to create a backup plan for the Database Service component and execute that plan. The Database Service component contains the configuration information for Secure API Manager and all of the APIs. If a disaster occurs, you can restore your data to the Database Service component and then redeploy the other Secure API Manager components to get your environment back up and running. We recommend that you cluster the Database Service component and create regular backups of each node in the cluster.

Secure API Manager stores the configuration information and the APIs in the `*appliance-name*/var/opt/microfocus/sapim/mount/mf-sapim-postgres/` directory on the Database Service component. Use your company's backup standard for Linux servers to create a backup of the following items on the Database Service appliance after you deploy it and it is up and functioning:

- The `*appliance-name*/var/opt/microfocus/sapim/mount/mf-sapim-postgres/` directory
- The ownership and file permission attributes for this directory and all of the directory's contents
- The `*appliance-name*/var/opt/microfocus/sapim/deployment-reference.json` file
- The `*appliance-name*/var/opt/microfocus/sapim/createPostgresContainer.sh` file

NetIQ recommends that you create a backup of the directory anytime the directory changes. It might be more than once a day. Whenever a developer creates a new API, Secure API Manager stores that API and all associated information in that directory. If your backup is not current and failure occurs, you will lose any APIs that are not in the backup. The other items in the list do not change after you deploy the Database Service appliance.

Setting a New Primary Database

When you cluster the Database Service component, by default the first node that you deploy is the primary database. In a disaster recovery situation where the primary database fails or you have to remove it from your system for any reason, you can specify a new primary database from the remaining database nodes. You can also remove other database nodes at the same time, if necessary. You can use any appliance to perform these steps, as long as it is not a database node that you plan to remove from your system.

To set a new primary database:

1. Log in to the appliance management console as `root`.

   `https://`*`ip-address-or-dns-name-appliance`*`:9443`

2. Click **Deployment Manager**.
3. On the **System** tab, locate the Database Service node that you want to make the primary database.
4. Click **Set As Primary**.
5. (Conditional) If you have not yet removed the current primary database from your system, select it from the list of databases that are available for removal.
6. (Optional) Select any other database nodes that you want to remove at this time.
7. Click **Set Primary Database**.

Secure API Manager updates the configuration to mark the selected appliance as the new primary database, and to remove all optionally selected appliances from the system. A redeployment automatically begins to apply the changes to the entire system, and the Status page opens so you can monitor the progress of the redeployment. Once the redeployment is complete, you have a functioning system with a new primary database.

Restoring a Failed Database Service

If the Database Service component fails but the other components are still running, you can power off the other components, redeploy the Database Service from backup, and then restart the other components in the proper order. We strongly recommend that you cluster the Database Service component and maintain regular backups. For more information, see the following topics:

- "Using High Availability and Load Balancing with Secure API Manager" on page 22
- "Preparing the Database Service Component for a Disaster" on page 61

IMPORTANT: Secure API Manager uses a master/master database setup to provide enhanced database performance due to high availability. This means that if a primary database fails and cannot simply be restarted, the secondary database must become the primary database. After that, you can add a secondary database to provide backup and high availability again.

In the event of database failure, the first step is to determine whether the *`appliance-name`*`/var/opt/microfocus/sapim/mount/mf-sapim-postgres/` directory is still available on a Database Service node.

- If the directory is available, you can follow the steps below to restore the Database Service from the backup, beginning with the primary database node.
- If you do not have this directory, it means that all databases failed without backup information. If all nodes have failed and you cannot access any database through the command line, you must completely remove and recreate the database cluster.

IMPORTANT: Assuming you have clustered and backed up the Database Service component, you must recover the primary database node before any other nodes. Complete the following steps to restore the primary database node, then restore the other database nodes.

To restore the Database Service from a backup:

1. Power off all of the other components and delete the Database Service component appliance from VMware.
2. Deploy an appliance for the Database Service component with the same networking configuration as it had before the failure. For more information, see "Deploying the Secure API Manager Appliances" on page 36.

   IMPORTANT: You must use the same network configuration for this appliance as it had before the failure, otherwise the restoration fails. The directory contains the databases for your system which includes the network settings.

3. On the appliance that will become the Database Service component, paste the backup directory *`appliance-name`*`/var/opt/microfocus/sapim/mount/mf-sapim-postgres/` to the new appliance.

   The Deployment Manager creates and populates this directory when you deploy a Database Service component. When the Deployment Manager detects that the directory already exists, it will not overwrite the information in the directory and it maintains all of the configuration information and APIs in the database.

4. On the appliance that will become the Database Service component, paste the following two files that you included in your backup plan to the *`appliance-name`*`/var/opt/microfocus/sapim` directory:
   - `deployment-reference.json`
   - `createPostgresContainer.sh`
5. Restart the Deployment Manager web application on the appliance that will become the Database Service component by issuing the following command at the appliance command prompt:

   ```
   systemctl restart vabase-jetty
   ```

6. From the command prompt in the *appliance-name*/var/opt/microfocus/sapim directory, execute the following two shell scripts in the order listed:

```
./createPostgresContainer.sh
./sapim-postgres-start.sh
```

7. Redeploy the Database Service component using the Deployment Manager, ensuring that you use the same database user name and password for the database. When saving the configuration, select **Save configuration and deploy only this appliance**. For more information, see "Deploying the Secure API Manager Components" on page 47.
8. (Conditional) If you have additional database nodes to redeploy, complete Step 2 through Step 7 for each additional node.
9. Power on the other Secure API Manager components in the proper order, ensuring that each component is up and communicating before starting the next component. For more information, see "Restarting Secure API Manager" in the *NetIQ Secure API Manager 1.1 Administration Guide*.

IMPORTANT: We strongly recommend that you cluster the Database Service component and make regular backups of the nodes in the cluster. If *all* Database Service nodes fail and you have not implemented a backup plan to capture a snapshot of the persistent database files, then the system is completely gone and you must recreate your entire system.

Enabling Email Notifications In Case of Failure of a Database Service Node

Secure API Manager provides the ability to send an email when a node in the Database Service cluster is down for longer than five minutes. The system still runs and everyone can still access the APIs and create APIs if a node in the Database Service cluster is down but eventually you can lose data. You should not run with one node down for an extended period of time.

Secure API Manager provides an SQL file that you configure for your environment to have the Database Service cluster send an email notification. You access the SQL file in the Customer Center (https://www.netiq.com/customercenter/app/home?execution=e1s1). For more information, see the Enabling Email Notifications for Secure API Manager 1.1 Technical Reference.

Recovering from a Failed Node in a Cluster

If one node in a cluster fails, Secure API Manager provides a process where you can redeploy a new node and have it join the system.

NOTE: The following steps apply only to non-database nodes. When you are redeploying a non-database failed node, you do not have to power down and restart the database nodes. For information about restoring a failed database node, see "Restoring a Failed Database Service" on page 62.

To remove and redeploy a failed node:

1. Ensure that you have completely removed the failed node from VMware.

2. Check the **SYSTEM** tab in the Deployment Manager to ensure that the remaining node or nodes in the cluster are up and communicating with the Database Service.
3. Redeploy a new appliance with the same IP address and DNS name as the failed node had before the failure. For more information, see "Deploying the Secure API Manager Appliances" on page 36.
4. Log in to the appliance management console on the new appliance as `root` with the new password you created when you redeployed the appliance.

   `https://`*`ip-address-or-dns-name-appliance`*`:9443`
5. Click **Deployment Manager**.
6. Select **Join**.
7. Specify the DNS name for the Database Service component and specify the database user name and password you created when you deployed your system.
8. Select **Join**.
9. Approve the certificate that the Deployment Manager displays or import a trusted root certificate for this appliance.
10. Click **Go To Deployment**.
11. Access the appropriate deployment page for this component, then specify the required information for this appliance.
12. On the last page, click **Save**.
13. Select **Save configuration and deploy only this appliance**.
14. On the **STATUS** tab, watch this node join the system. The Deployment Manager adds the configuration information for this new node to each component in the system.

Restoring a Failed Secure API Manager System

If a disaster occurs and you lose all of the Secure API Manager components or the entire system becomes corrupted, you can restore the entire system as long as you have a copy of the backup directory from the Database Service component. For more information, see "Preparing the Database Service Component for a Disaster" on page 61.

If you have clustered the Database Service component and have a backup for each Database Service appliance node, in the event that you need to restore your entire Secure API Manager system, restore all of the databases from the backups before deploying the other components.

WARNING: You must use the same networking configuration settings as you had in the failed system. If you change the networking configuration settings, the restoration fails.

To restore a system with the backup file from the Database Service:

1. Delete all of the appliances from VMware and the load balancers.
2. Deploy the appropriate number of appliances that were in your environment, making sure that you use the same network configuration settings. For more information, see "Deploying the Secure API Manager Appliances" on page 36.

3 On the appliance where you installed the new Database Service component, replace the `*appliance-name*/var/opt/microfocus/sapim/mount/mf-sapim-postgres/persistent` directory that is currently on the appliance with the backed-up directory from your original Database Service component.

4 Deploy the Database Service component using the Deployment Manager, ensuring that you use the same database user name and password for the database.

5 Deploy the remaining components. For more information, see "Deploying the Secure API Manager Components" on page 47.

A Documentation Updates

The following section contains a list of changes to the documentation.

July 2019

Location	Change
"Enterprise Deployment Scenario On-Premises" on page 20	Changed the deployment graphic and updated the text in this section to match the new graphic.