Setup Guide

Access Manager Appliance 3.2 SP2

June 2013

Legal Notice

About This Guide

This guide is intended to help you understand and set up a basic Access Manager Appliance configuration.

IMPORTANT: To avoid configuration errors, it is strongly recommended that you closely follow the steps outlined in this document during your initial Access Manager Appliance setup.

Not all Access Manager Appliance functionality and administrative tasks are discussed here. After you are familiar with Access Manager Appliance and the steps in this section, you can use the *NetIQ Access Manager Appliance 3.2 SP2 Identity Server Guide* and the *NetIQ Access Manager Appliance 3.2 SP2 Access Gateway Guide* as the sources for additional or advanced configuration.

Audience

This guide is intended for Access Manager Appliance administrators. It is assumed that you have knowledge of evolving Internet protocols, such as:

- Extensible Markup Language (XML)
- Simple Object Access Protocol (SOAP)
- Security Assertion Markup Language (SAML)
- Public Key Infrastructure (PKI) digital signature concepts and Internet security
- Secure Socket Layer/Transport Layer Security (SSL/TLS)
- Hypertext Transfer Protocol (HTTP and HTTPS)
- Uniform Resource Identifiers (URIs)
- Domain Name System (DNS)
- Web Services Description Language (WSDL)

Feedback

We want to hear your comments and suggestions about this manual and the other documentation included with this product. Please use the User Comments feature at the bottom of each page of the online documentation.

Documentation Updates

For the most recent version of the *Setup Guide*, visit the NetIQ Access Manager Documentation Web site (https://www.netiq.com/documentation/novellaccessmanager32/).

Additional Documentation

- *NetIQ Access Manager Appliance 3.2 SP2 Administration Console Guide*
- *NetIQ Access Manager Appliance 3.2 SP2 Identity Server Guide*
- *NetIQ Access Manager Appliance 3.2 SP2 Access Gateway Guide*
- *NetIQ Access Manager Appliance 3.2 SP2 SSL VPN Server Guide*
- *NetIQ Access Manager Appliance 3.2 SP2 Policy Guide*

NOTE: Contact namsdk@netiq.com for any query related to Access Manager SDK.

Contents

1 Setting Up a Basic Access Manager Appliance Configuration

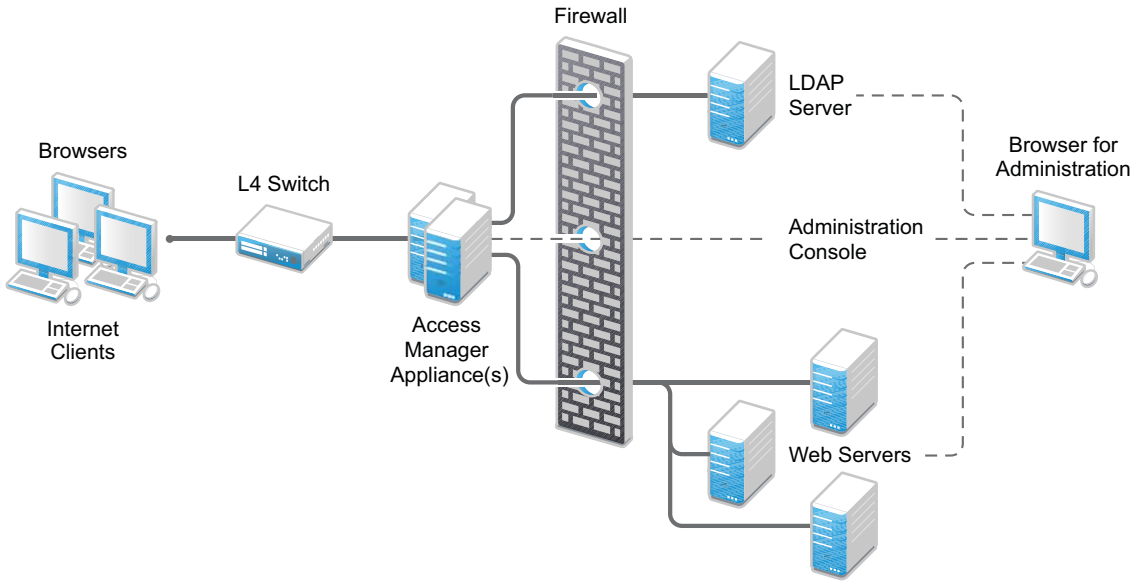
The initial setup consists of installing NetIQ Access Manager Appliance. You must set up the User Stores for Identity Server and configure the Access Gateway to protect resources running on an HTTP Web server.

This tutorial describes the following topics and tasks:

1.1 Understanding Access Manager Appliance Configuration

The following figure illustrates the components and process flow that make up a basic configuration.

Figure 1-1 *Basic Process Flow*

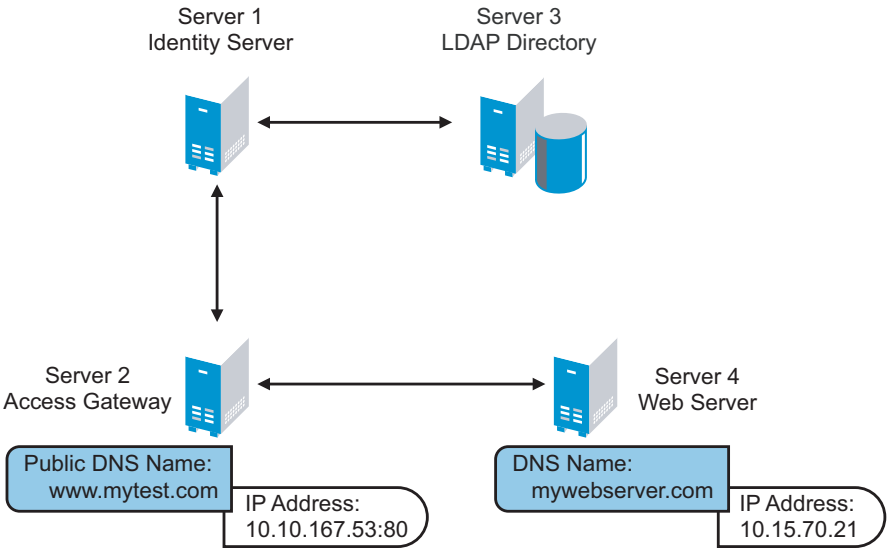


1. The user sends a request to the Access Gateway for access to a protected resource.
2. The Access Gateway redirects the user to the Identity Server, which prompts the user for a username and password.

3. The Identity Server verifies the username and password against an LDAP directory user store (eDirectory, Active Directory, or Sun ONE).
4. The Identity Server returns an authentication artifact to the Access Gateway through the browser in a query string.
5. The Access Gateway retrieves the user's credentials from the Identity Server through the SOAP channel in the form of a SOAP message.
6. The Access Gateway injects the basic authentication information into the HTTP header.
7. The Web server validates the authentication information and returns the requested Web page.

You configure the Access Manager Appliance so that a user can access a resource on a Web server whose name and address are hidden from the user. This basic configuration sets up communication between the following four servers:

Figure 1-2 *Basic Configuration*



Although other configurations are possible, this section explains the configuration tasks for this basic Access Manager Appliance configuration. This section explains how to set up communication using HTTP. For HTTPS over SSL, see Chapter 2, "Enabling SSL Communication," on page 21.

1.2 Prerequisites for Setup

The following prerequisites are for setting up a basic Access Manager Appliance configuration:

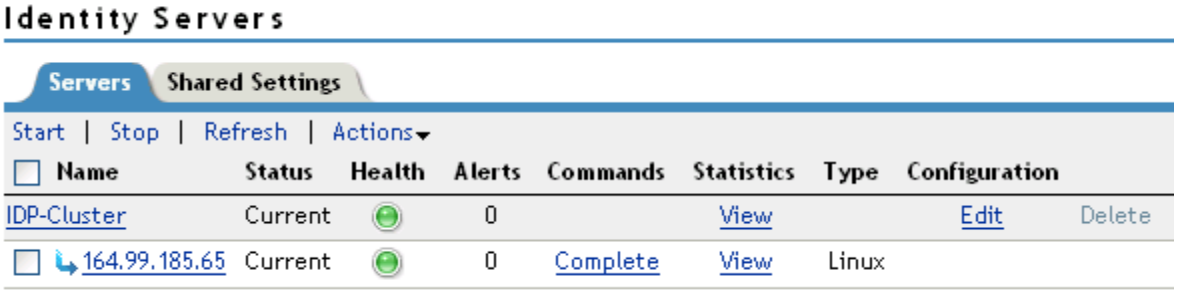
- ❒ An installed Access Manager Appliance. See "Installing the Access Manager Appliance" in the *NetIQ Access Manager Appliance 3.2 SP2 IR1 Installation Guide*.
- ❒ An LDAP directory store with a test user added. This store can be eDirectory, Active Directory, or Sun ONE.
- ❒ A DNS server or modified `host` files to resolve DNS names and provide reverse lookups. For information about which `host` files need to be modified, see Configuring Name Resolution (https://www.netiq.com/documentation/netiqaccessmanager32/basicconfig/data/bbjmalj.html#bajlym3).

- ❒ A Web server (IIS or Apache). The Web server should have three directories with three HTML pages. The first directory (`public`) should contain a page (such as index.html) for public access. This page needs to provide two links:
  - A link to a page in the `protected` directory. You will configure the Access Gateway to require authentication before allowing access to this page. You do not need to configure the Web server to protect this page.
  - A link to a page in the `basic` directory. You should already have configured your Web server to require basic authentication before allowing access to this page. See your Web Server documentation for instructions on setting up basic authentication. (This type of access is optional, but explained because it is fairly common.)

  If you do not have a Web server that you can use for this type of access, you might prefer to configure Access Manager Appliance for the sample Web pages we provide. See (https://www.netiq.com/documentation/netiqaccessmanager32/basicconfig/data/bayxa4y.html).
- ❒ A client workstation with a browser with browser pop-ups enabled.

1.3 Setting up User Stores for Identity Server Configuration

After you log in to the Administration Console, click **Devices** > **Identity Servers**. The system displays the installed server, as shown in the following example:

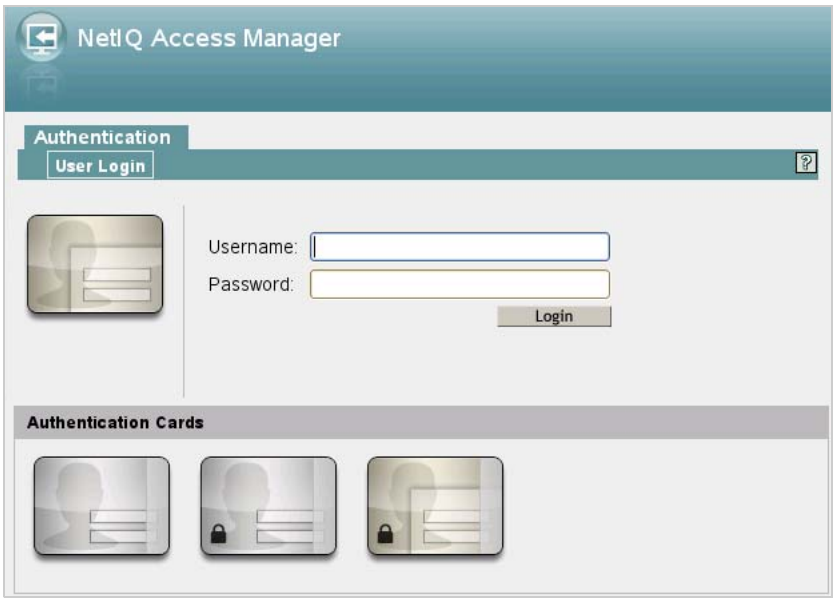


At this point the Identity Server is in configured state and is functional with Default User Store.

To configure the user store, you need the following information:

- The IP address of an LDAP directory (user store). The LDAP directory is used to authenticate users. The trusted root certificate of the user store is imported to provide secure communication between the Identity Server and the user store.
- The distinguished name and password of the administrator of the LDAP user store.

1. Configure the User Store. For information about configuring User Store, see "Configuring the User Store " in the *NetIQ Access Manager Appliance 3.2 SP2 Identity Server Guide*.
2. (Optional) Verify the configuration:
   - 2a. In a browser, enter the Base URL of the NetIQ Access Manager Appliance. Click the Sample Application Link. You will be redirected to the Login Page.



2b Log in using the credentials of a user in the LDAP server.

2c (Conditional) If the URL returns an error rather than displaying a login page, verify the following:

- The browser machine can resolve the DNS name of the Identity Server.
- The browser machine can access the port.

1.4 Configuring the Access Gateway

The basic Access Gateway configuration procedures have been divided into the following tasks:

- Section 1.4.1, “Configuring a Reverse Proxy,” on page 12
- Section 1.4.2, “Configuring a Public Protected Resource,” on page 15

1.4.1 Configuring a Reverse Proxy

You protect your Web services by creating a reverse proxy. A reverse proxy acts as the front end to your Web servers in your DMZ or on your intranet, and off-loads frequent requests, thereby freeing up bandwidth and Web server connections. It also increases security because the IP addresses and DNS names of your Web servers are hidden from the Internet. A reverse proxy can be configured to protect one or more proxy services. To configure the Access Gateway, you can create a new configuration as described in this section.

To create a reverse proxy, you must create at least one proxy service with a protected resource. You must supply a name for each of these components. Reverse proxy names and proxy service names must be unique to the Access Gateway because they are configured for global services such as IP addresses and TCP ports. For example, if you have a reverse proxy named `products` and another reverse proxy named `library`, only one of these reverse proxies can have a proxy service named `corporate`.

You can also modify the existing default NAM-RP to match your requirement. The Access Manager Appliance has a default SSL-enabled reverse proxy (NAM-RP). The reverse proxy is associated with a self-signed certificate, which is created during installation of the primary Access Manager Appliance. To modify the default NAM-RP, click **Devices** > **Access Gateways** > **Edit** > **NAM-RP** in the

Administration Console. The default proxy service is NAM-Service. You cannot delete this proxy service and base service. You can modify, enable, disable, rename, and delete the Path-Based Multi-Homing (PBMH), which is created under this proxy service. You can create a new PBMH or Domain-Based Multi-Homing (DBMH) under NAM-service. You can also create a new protected resource, which you can assign it to the newly created PBMH or DBMH. The protected resource, which are not greyed out can also be used to add, delete, modify, enable, and disable paths.

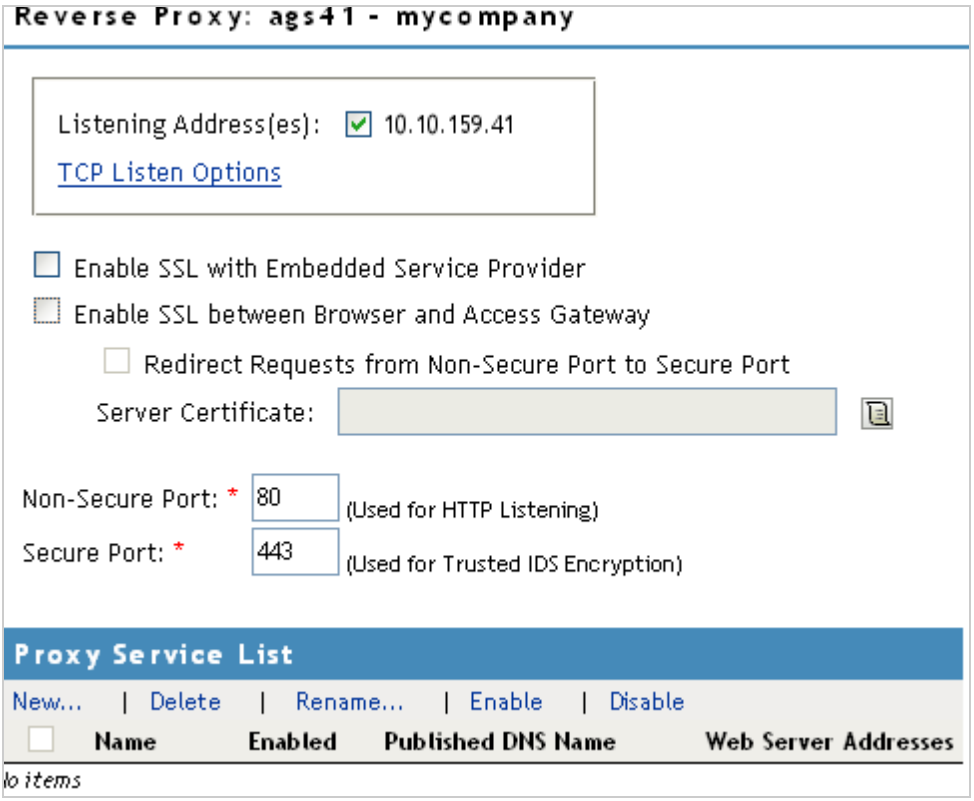
Protected resource names need to be unique to the proxy service, but they don't need to be unique to the Access Gateway because they are always accessed through their proxy service. For example, if you have a proxy service named `account` and a proxy service named `sales`, they both can have a protected resource named `public`.

What You Need To Know	Example	Your Value
DNS name of the Access Gateway	mytest.com	_____
Web server information		
IP address	10.15.70.21	_____
DNS name	mywebserver.com	_____
Names you need to create		
Reverse proxy name	mycompany	_____
Proxy service name	company	_____
Protected resource name	public	_____

This first reverse proxy is used for authentication. You need to configure the proxy service to use the DNS name of the Access Gateway as its **Published DNS Name**, and the Web server and the resource on that Web server need to point to the page you want displayed to the users when they first access your Web site. You can use Access Gateway configuration options to allow this first page to be a public site with no authentication required until the users access the links on the page, or you can require authentication on this first page. The following configuration steps have you first configure the protected resource as a public resource, then you modify the configuration to require authentication.

1. In the Administration Console, click **Devices** > **Access Gateways**, then click **Edit** > **Reverse Proxy / Authentication**.
2. In the **Reverse Proxy List**, click **New**, specify a display name for the reverse proxy, then click **OK**.

The Reverse Proxy configuration page appears.



3 Enable a listening address.

Listening Address(es): A list of available IP addresses. If the server has only one IP address, only one is displayed and it is automatically selected. If the server has multiple addresses, you can select one or more IP addresses to enable. You must enable at least one address by selecting its check box.

TCP Listen Options: Options for configuring how requests are handled. You cannot set up the listening options until you create a proxy service.

4 Ignore the SSL configuration options.

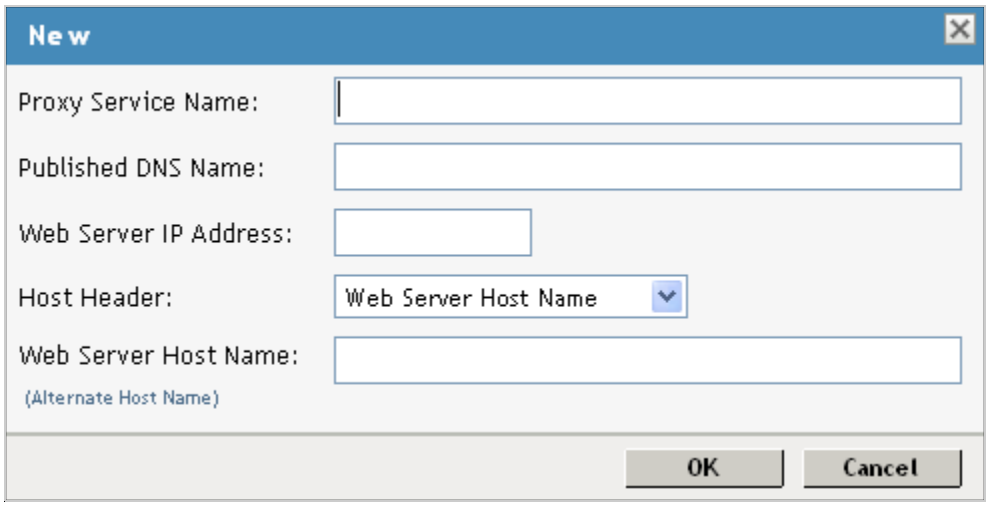
This basic configuration does not set up SSL. For SSL information, see Chapter 2, "Enabling SSL Communication," on page 21.

5 Configure a listening port.

Non-Secure Port: Select 80, which is the default port for HTTP.

Secure Port: This is the HTTPS listening port. This port is unused and cannot be configured until you enable SSL.

6 In the **Proxy Service List**, click **New**.



7 Fill in the fields.

Proxy Service Name: A display name for the proxy service.

Published DNS Name: The DNS name you want the public to use to access your site. For this first proxy server, the DNS name must resolve to the Access Gateway IP address that you selected as the listening address. For the example in Figure 1-2 on page 10, this name would be www.mytest.com.

Web Server IP Address: The IP address of your Web server. This is usually a Web server with content that you want to share with authorized users and protect from all others. In Figure 1-2 on page 10, this is Server 4, whose IP address is 10.15.70.21.

Host Header: The name you want sent in the HTTP header to the Web server. This can be either the Published DNS Name (the **Forward Received Host Name** option) or the DNS name of the Web Server (the **Web Server Host Name** option).

Web Server Host Name: The DNS name that the Access Gateway should forward to the Web server. This option is not available if you selected **Forward Received Host Name** for the **Host Header** option. The name you use depends upon how you have set up the Web server. If your Web server has been configured to verify that the host name in the header matches its name, you need to specify that name here. In Figure 1-2 on page 10 the Web Server Host Name is mywebserver.com.

8 Click **OK**.

9 Continue with Section 1.4.2, “Configuring a Public Protected Resource,” on page 15.

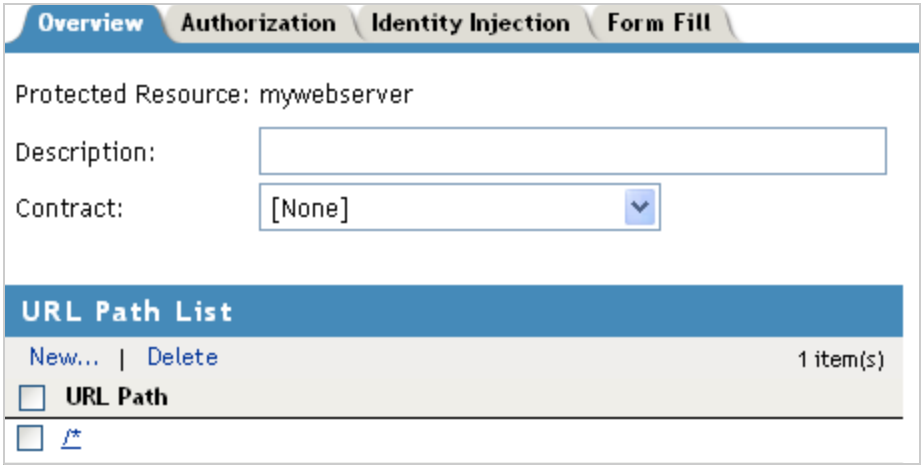
1.4.2 Configuring a Public Protected Resource

The first protected resource in this configuration tutorial is configured to be a public resource.

1 In the **Proxy Service List**, click **[Name of Proxy Service]** > **Protected Resources**.

2 In the **Protected Resource List**, click **New**.

3 Specify a display name for the protected resource, then click **OK**.



4 (Optional) Specify a description for the protected resource.

5 In the **Contract** field, select **None**.

The **Contract** field must be set to **None**. This is what makes this resource a public resource.

6 Configure the **URL Path List**.

The default path is `/*`, which allows access to everything on the Web server. Modify this if you need to restrict access to a specific directory on your Web server.

- To delete the default path, select the check box next to the path, then click **Delete**.
- To edit a path in the list, click the path, modify it, then click **OK**.
- To add a path, click **New**, specify the path, then click **OK**. For example, to allow access to the pages in the public directory on the Web server, specify the following path:

  ```
  /public/*
  ```

7 Click **OK**.

8 In the **Protected Resource List**, verify that the protected resource you created is enabled, then click OK.

9 Click the **Devices** > **Access Gateways**.

10 To apply the changes, click **Update** > **OK**.

Until this step, nothing has been permanently saved or applied. The **Update** status pushes the configuration to the server and writes the configuration to the configuration data store. When the update has completed successfully, the server returns the status of **Current**.

To save the changes to the configuration store without applying them, do not click **Update**. Instead, click **Edit**. If you have pending configuration settings, the **OK** button is active, and the configuration page indicates which services will be updated. Click **OK** to write these changes to the configuration store. The changes are not applied until you click **Update** on the Access Gateways page.

11 To update the Identity Server to establish the trust relationship with the Access Gateway, click **Devices** > **Identity Servers** > **Update**, then click **OK**.

Wait until the **Command** status is **Complete** and the **Health** status is green.

12 (Optional). To test this configuration from a client browser, enter the published DNS name as the URL in the browser. For the example illustrated in Figure 1-2 on page 10, you would enter the following URL:

```
http://www.mytest.com
```

This should resolve to the published DNS name you specified in Step 7 on page 15, and the user should be connected to the Web server through the Access Gateway.

1.5 Setting Up an Identity Injection Policy

The Access Gateway lets you retrieve information from your LDAP directory and inject the information into HTML headers, query strings, or basic authentication headers. The Access Gateway can then send this information to the back-end Web servers. Access Manager calls this technology Identity Injection. iChain calls it Object Level Access Control (OLAC). This is one of the features within Access Manager that enables single sign-on. The user is prompted once for the login credentials, and Access Manager then supplies them for the resources you have configured for Identity Injection.

This section explains how to set up an Identity Injection policy for basic authentication. This policy is assigned to the third directory on your Web server, which is the `basic` directory that your Web server has been configured to require basic authentication before allowing access.

1 In the Administration Console, click **Devices** > **Access Gateways** > **Edit** > **[Reverse Proxy Name]** > **[Proxy Service Name]** > **Protected Resources** > **New**.

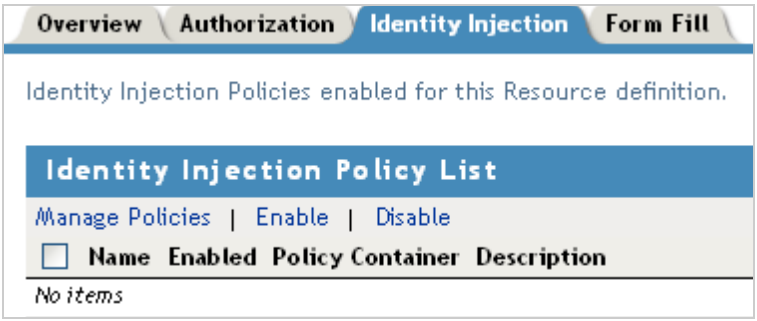
2 Configure the resource for the `basic` directory as described in Section 1.2, "Prerequisites for Setup," on page 10:

2a For the contract, select **Name/Password - Basic** or **Name/Password - Form**.

2b For the URL path, enter the path to the basic directory (`/basic/*`).

2c Click **OK**.

3 Click **[Protected Resource Name]** > **Identity Injection**.



On a new installation, the list is empty because no policies have been created.

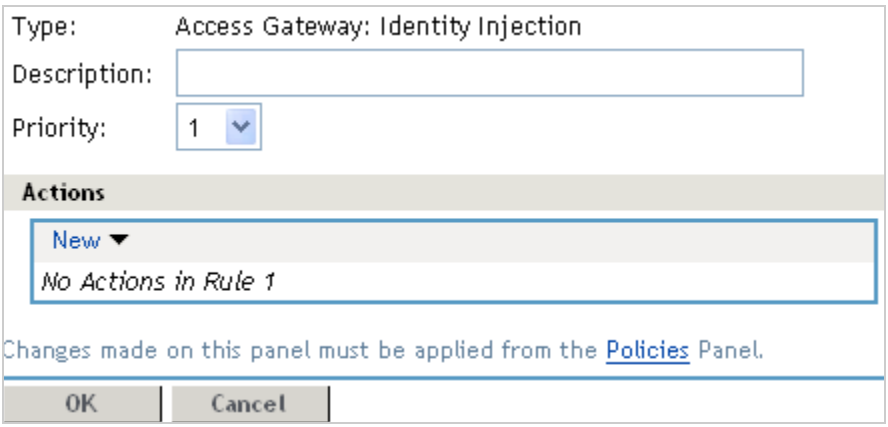
4 In the **Identity Injection Policy List** section, click **Manage Policies**.

5 In the **Policy List** section, click **New,** then specify values for the following fields:

Name: Specify a name for the Identity Injection policy.

Type: Select **Access Gateway: Identity Injection**.

6 Click **OK**.



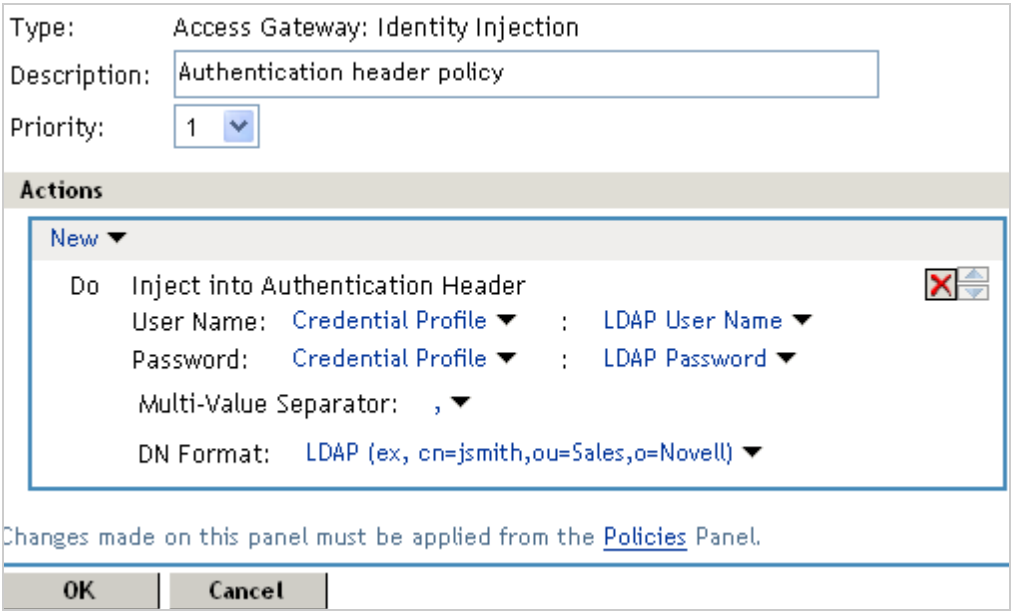
7 (Optional) Specify a description for the policy.

8 In the **Actions** section, click **New** > **Inject into Authentication Header**.

9 Set up the policy for **User Name** and **Password**:

- For **User Name**, select **Credential Profile** and **LDAP Credentials: LDAP User Name**. This injects the value of the cn attribute into the header.
- For **Password**, select **Credential Profile** and **LDAP Credentials: LDAP Password**.

The policy should look similar to the following:



10 Click **OK** twice, then click **Apply Changes**.

11 Click **Close**.

12 Select the new Identity Injection policy, then click **Enable**.

13 To save the changes to browser cache, click **OK**.

14 To apply your changes, click **Devices** > **Access Gateways**, then click **Update** > **OK**.

15 To test this configuration from a client browser, enter the published DNS name as the URL in the browser. Click the link to the page that uses basic authentication.

   You are prompted to log in. If you have set up Web applications on your Web server that require login, any additional login prompts are hidden from the user and are handled by the identity injection system.

For an example of how Identity Injection policies can be used for single sign-on to the Identity Manager User Application, see "Configuring Access Manager for UserApp and SAML" (http://www.novell.com/coolsolutions/appnote/19981.html).

2 Enabling SSL Communication

NetIQ Access Manager Appliance enables SSL communication with the Default Reverse Proxy and the Identity Server, using a self signed certificate.

You can configure the Access Gateway to use SSL in its connections to the browsers, and to its Web servers.

2.1 Using Access Manager Certificates

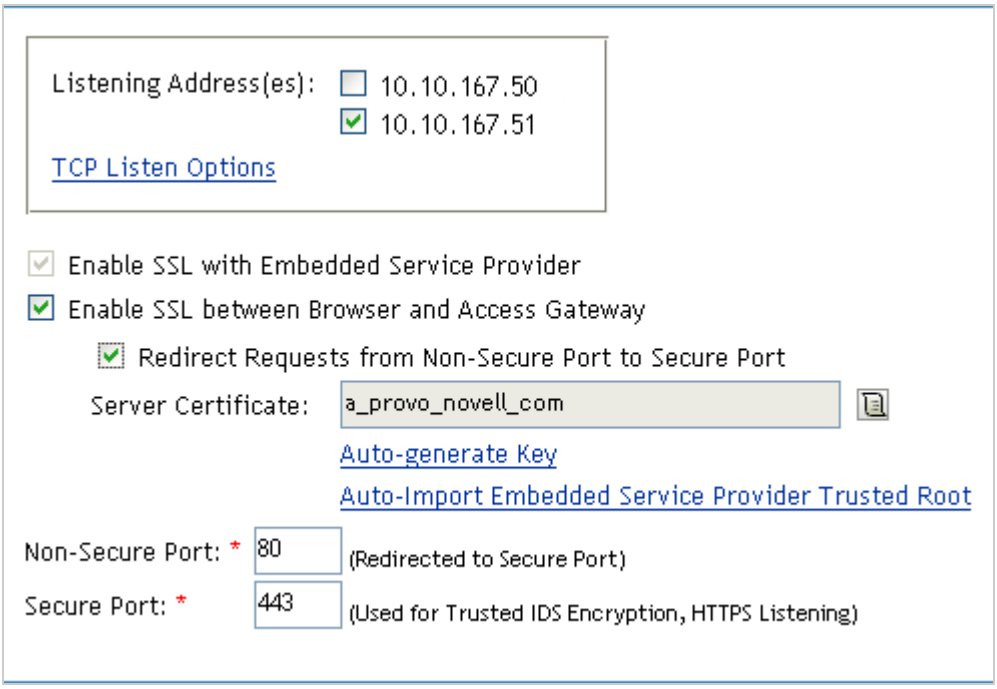
By default, all Access Manager Appliance components (Identity Server, Access Gateway, and SSL VPN) trust the local CA. However, the browsers are not set up to trust the Access Manager CA. You need to import the public key of the trusted root certificate (configCA) into the browsers to establish the trust.

2.1.1 Configuring the Access Gateway for SSL

This section describes how to set up SSL for the Access Gateway communication channels:

Configuring SSL Communication with the Browsers and the Access Gateway

1 In the Administration Console, click **Devices** > **Access Gateways** > **Edit** > **[Name of Reverse Proxy]**.



2 To configure the reverse proxy for SSL, fill in the following fields:

Enable SSL with Embedded Service Provider: Select this option to encrypt the data exchanged for authentication (the communication channel between the Identity Server and the Access Gateway). This option is only available for the reverse proxy that has been assigned to perform authentication.

If you enable SSL between the browsers and the Access Gateway, this option is automatically selected for you. You can enable SSL with the embedded service provider without enabling SSL between the Access Gateway and the browsers. This allows the authentication and identity information that the Access Gateway and the Identity Server exchange to use a secure channel, but allows the Access Gateways to use non-secure channels with the browsers and the Web servers. This saves processing overhead if the data on the Web servers is not sensitive.

Enable SSL between Browser and Access Gateway: Select this option to require SSL connections between your clients and the Access Gateway. SSL must be configured between the browsers and the Access Gateway before you can configure SSL between the Access Gateway and the Web servers. For this process, see "Enabling SSL between the Reverse Proxy and Its Web Servers" on page 24.

Redirect Requests from Non-Secure Port to Secure Port: Determines whether browsers are redirected to the secure port and allowed to establish an SSL connection. If this option is not selected, browsers that connect to the non-secure port are denied service.

3 Generate a certificate key by using the Access Manager CA:

3a Click **Auto-generate Key**, then click **OK** twice.

3b On the Select Certificate page, make sure the certificate is selected, then click **OK**.

The generated certificate appears in the **Server Certificate** text box.

4 Configure the ports for SSL:

Non-Secure Port: Specifies the port on which to listen for HTTP requests. The default port for HTTP is 80. If you have selected the **Redirect Requests from Non-Secure Port to Secure Port** option, requests sent to this port are redirected to the secure port. If the browser can establish an SSL connection, the session continues on the secure port. If the browser cannot establish an SSL connection, the session is terminated.

Secure Port: Specifies the port on which to listen for HTTPS requests (which is usually 443). This port needs to match the configuration for SSL. If SSL is enabled, this port is used for all communication with the browsers. The listening address and port combination must not match any combination you have configured for another reverse proxy or tunnel.

5 In the **Proxy Service List**, click **[Name of Proxy Service]** > **Protected Resources**.

6 In the **Protected Resource List**, change the Authentication Procedure from an HTTP contract to an HTTPS contract.

For example, if a protected resource is using the **Name/Password - Basic** contract, click the name and change it to the **Name/Password - Form**, the **Secure Name/Password - Basic** or the **Secure Name/Password - Form** contract. Then click **OK**.

The **Name/Password - Form** contract is capable of using either HTTP or HTTPS.

To enable single sign-on, select the same contract for all the protected resources.

7 Click the **Configuration Panel** link near the bottom of the page, then in the confirmation box, click **OK**.

8 On the Server Configuration page, click **Reverse Proxy / Authentication**.

9 In the **Embedded Service Provider** section, click **Auto-Import Identity Server Configuration Trusted Root**, click **OK**, specify an alias, click **OK** twice, then click **Close**.

This option imports the public key of the Identity Server into the trust store of the embedded service provider. This sets up a trusted SSL relationship between the embedded service provider and the Identity Server.

The configCA public key certificate of the Access Manager CA is automatically added to the ESP Trust Store. If you are using Access Manager CA certificates for the Identity Server, you do not need to import the configCA certificate unless someone has deleted it from this trust store.

10 Click **Configuration Panel**, then in the confirmation box, click **OK**.

11 On the Server Configuration page, click **OK**.

12 On the Access Gateways page, click **Update** > **OK**.

13 Update the Identity Server so that it uses the new SSL configuration. Click **Devices** > **Identity Servers**, then click **Update** > **OK**.

14 Verify that the trusted relationship between the Identity Server and the Access Gateway has been reestablished:

14a Enter the URL to a protected resource on the Access Gateway. For example, enter

```
https://www.mytest.com
```

14b Complete one of the following:

- If you are prompted for login credentials, enter them. The trusted relationship has been reestablished.
- If you receive a 100101043 or 100101044 error, the trusted relationship has not been established. For information about solving this problem, see "Troubleshooting 100101043 and 100101044 Liberty Metadata Load Errors" in the *NetIQ Access Manager Appliance 3.2 SP2 Identity Server Guide*.

Enabling SSL between the Reverse Proxy and Its Web Servers

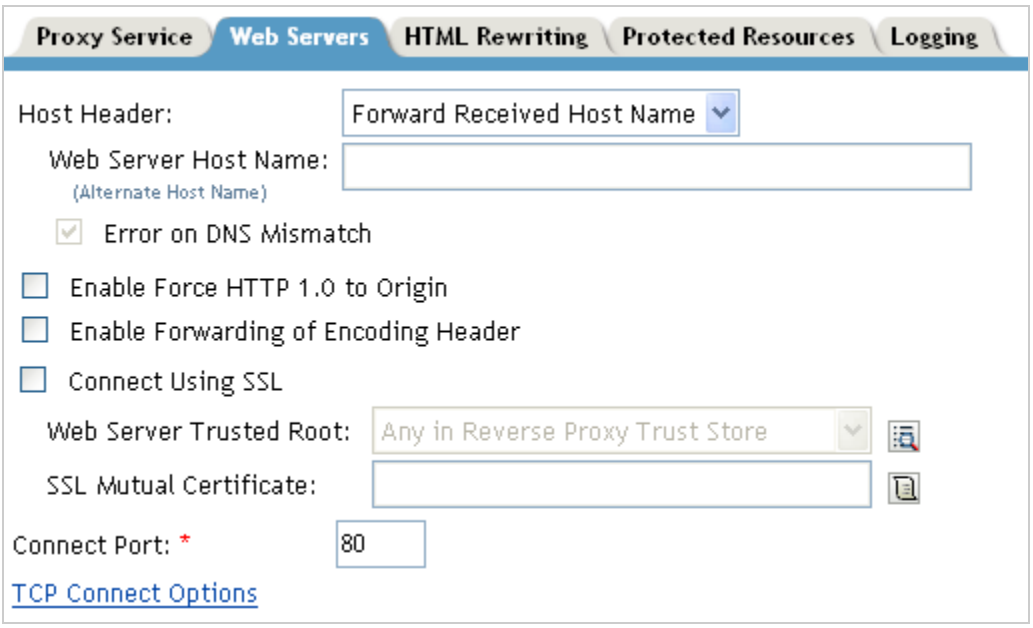
To enable SSL between the reverse proxy and the Web servers, you must have already performed the following tasks:

- ❒ Enabled SSL between the Access Gateway and the browsers. See Section 1.4.1, "Configuring a Reverse Proxy," on page 12 and select the **Enable SSL between Browser and Access Gateway** field.
- ❒ Enabled SSL on the Web server. See your Web server documentation.

If you have completed these tasks:

1. In the Administration Console, click **Devices** > **Access Gateways** > **Edit** > **[Name of Reverse Proxy]** > **[Name of Proxy Service]** > **Web Servers**.

   The Web Servers configuration page appears.



2. To configure SSL, select **Connect Using SSL**.

   This option is not available if you have not set up SSL between the browsers and the Access Gateway. See Section 1.4.1, "Configuring a Reverse Proxy," on page 12 and select the **Enable SSL between Browser and Access Gateway** field.

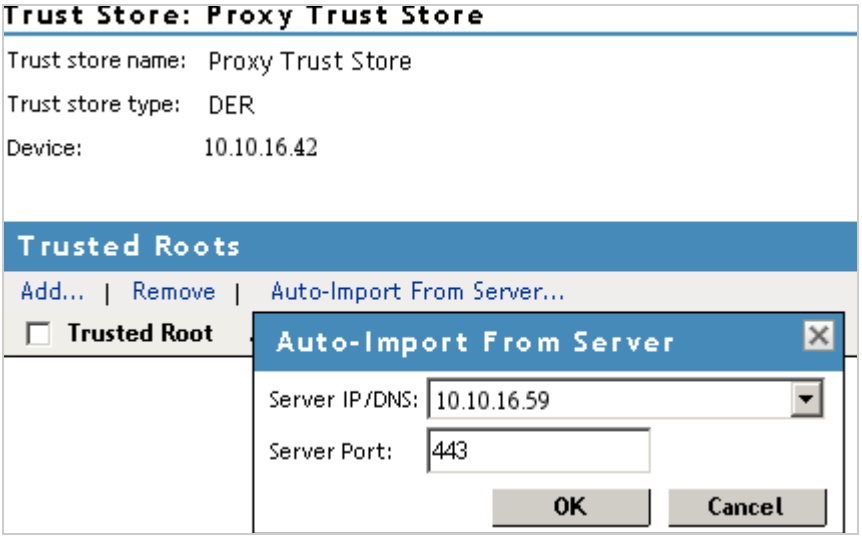
3. In the **Connect Port** field, specify the port that your Web server uses for SSL communication.

4. Configure how you want the certificate verified. The Access Gateway supports different options. Select one of the following:

   - **Do not verify:** Select this option if you do not want to verify the **Web Server Trusted Root** certificate. Continue with Step 10.
   - To verify the certificate authority of the Web server certificate, select **Any in Reverse Proxy Trust Store**. When this option is selected, the public certificate of the certificate authority must be added to the proxy trust store.

IMPORTANT: For an Access Gateway Service, this option is a global option. If you select this option for one proxy service, all proxy services on an Access Gateway Service are flagged to verify the public certificate. This verification is done even when other proxy services are set to **Do not verify**.

5 Click the **Manage Reverse Proxy Trust Store** icon. The auto import screen appears.



6 Ensure that the IP address of the Web server and the port match your Web server configuration.

If these values are wrong, you have entered them incorrectly on the Web server page. Click **Cancel** and reconfigure them before continuing.

7 Click **OK**.

Wait while the Access Gateway retrieves the server certificate, the root CA certificate, and any CA certificates from a chain from the Web server.

8 Specify an alias, then click **OK**.

All the displayed certificates are added to the trust store.

9 Click **Close**.

10 (Optional) For mutual authentication:

10a Select the certificate. Click the **Select Certificate** icon, select the certificate you created for the reverse proxy, then click **OK**.

10b Import the trusted root certificate of the CA that signed the proxy service's certificate to the Web servers assigned to this proxy service.

See your Web server documentation for instructions.

11 Click **Configuration Panel**, then click **OK**.

12 On the **Configuration** page, click **OK**.

13 On the **Access Gateways** page, click **Update**.

14 (Optional). Test this configuration from a client browser:

14a Enter the published DNS name as the URL in the browser.

14b Click the links that require authentication for access.

2.2 Using Externally Signed Certificates

When the Identity Server is configured to use an SSL certificate that is signed externally, the trusted store of the embedded service provider for each component must be configured to trust this new CA. The browsers that are used to authenticate to the Identity Server must be configured to trust the CA that created the certificate for the Identity Server. If you obtain a certificate from a well-known external CA, most browsers are already configured to trust certificates from well-known CAs.

The following procedures explain how to use certificates signed by an external Certificate Authority.

- Section 2.2.1, "Obtaining Externally Signed Certificates," on page 26
- Section 2.2.2, "Configuring the Access Gateway to Use an Externally Signed Certificate," on page 28

2.2.1 Obtaining Externally Signed Certificates

The following sections explain how to create certificate signing requests for the Identity Server and Access Gateway, how to use the requests to obtain signed certificates, then how to import the signed certificates and the root certificate of the Certificate Authority into Access Manager Appliance.

- "Creating the Certificate Signing Request" on page 26
- "Getting a Signed Certificate" on page 27
- "Importing the Signed Certificates and Root Certificate" on page 27

Creating the Certificate Signing Request

You need to create two certificate signing requests: one for the Identity Server and one for the Access Gateway. The **Certificate name** and the **Common name** need to be different, but the other values can be the same.

What you need to know or create	Example	Your Value
Certificate name	ipda_test or lag_test	_____ _____
Certificate Subject Fields:		
Common name	ipda.test.novell.com or lag.test.novell.com	_____ _____
Organizational unit	novell	_____
Organization	test	_____
City or town	Provo	_____
State or province	UTAH	_____
Country	US	

To create a signing request for the Identity Server:

1. In the Administration Console, click **Security** > **Certificates** > **New**.
2. Select the **Use External certificate authority** option.

3 Fill the following fields:

Certificate name: `idpa_test`

Signature algorithm: Accept the default.

Valid from: Accept the default.

Months valid: Accept the default.

Key size: Accept the default.

4 Click the **Edit** icon on the **Subject** line.

5 Fill in the following fields:

Common name: `idpa.test.novell.com`

Organizational unit: `novell`

Organization: `test`

City or town: `Provo`

State or province: `UTAH`

Country: `US`

6 Click **OK** twice, then click the name of the certificate.

7 Click **Export CSR**.

The signing request is saved to a file.

8 Repeat Step 1 through Step 7 to create a signing request for the Access Gateway.

Getting a Signed Certificate

You can send the certificate signing request to a certificate authority and wait for the CA to return a signed certificate or you can use a trial certificate for testing while you wait for the official certificate. Companies such as VeriSign offer trial signed certificates for testing.

Modify the following instructions for the CA you have selected to sign your certificates:

1 Set up an account with a certificate authority and select the free trial option.

2 Open your certificate signing request for the Identity Server in a text editor.

3 Copy and paste the text of the certificate request into the appropriate box for a trial certificate.

4 If CA requires that you select a server platform, select eDirectory if available. If eDirectory is not a choice, select unknown or server not listed.

5 Click **Next**, then copy the signed certificate and paste it into a new text file or at the bottom of the signing request file.

6 Click **Back**, and repeat Step 2 through Step 5 for the Access Gateway.

7 Follow the instructions of the vendor to download the root certificate of the Certificate Authority and any intermediate CA certificates.

Importing the Signed Certificates and Root Certificate

The following steps explain how to imported the signed certificates and the trust root into the Administration Console so that they are available to be assigned to key stores and trusted root stores.

1 In the Administration Console, click **Access Manager** > **Certificates** > **Trusted Roots**.

2 Click **Import**, then specify a name for the root certificate.

3. Either click **Browse** and locate the root certificate file or select **Certificate data text** and paste the certificate in the text box.
4. Click **OK**.

   The trusted root is added and is now available to add to trusted root stores.
5. (Conditional) Repeat Step 2 through Step 4 for any intermediate CA certificates.
6. In a text editor, open the signed certificate for the Identity Server.
7. In the Administration Console, click **Access Manager** > **Certificates**, then click the name of certificate signing request for the Identity Server.
8. Click **Import Signed Certificate**, then select **Certificate data text (PEM/Based64)**.
9. Paste the text for the signed certificate into the data text box. Copy everything from

   ```
   -----BEGIN CERTIFICATE-----
   ```

   through

   ```
   -----END CERTIFICATE-----
   ```
10. Click **Add trusted root**, then either click **Browse** and locate the root certificate file or select **Certificate data text** and paste the certificate in the text box.
11. (Conditional) For any intermediate CA certificates, click **Add intermediate certificate**, then either click **Browse** and locate the intermediate certificate file or select **Certificate data text** and paste the certificate in the text box.
12. Click **OK**.

    The certificate is now available to be assigned to the keystore of a device.

    If the certificate fails to import and you receive an error, it is probably missing a trusted root certificate in a chain of trusted roots. To determine whether this is the problem, see "Resolving a -1226 PKI Error " and "Importing an External Certificate Key Pair" in the *NetIQ Access Manager Appliance 3.2 SP2 Administration Console Guide*.
13. Repeat Step 6 through Step 12 for the Access Gateway certificate.

2.2.2 Configuring the Access Gateway to Use an Externally Signed Certificate

This section explains how to enable SSL communication between the Access Gateway and the Identity Server and between the Access Gateway and the browsers.

1. In the Administration Console, click **Devices** > **Access Gateways** > **Edit** > **[Name of Reverse Proxy]**.
2. Select **Enable SSL between Browser and Access Gateway**.
3. In the **Server Certificate** line, click the **Browse** icon.
4. Select the Access Gateway certificate, then click **OK**.

   IMPORTANT: If the external certificate authority writes the DN in reverse order (the cn element comes first rather than last), you receive an error message that the subject name does not contain the cn of the device. You can ignore this warning, if the order of the DN elements is the cause.

   ---
5. Specify an **Alias** for the certificate, then click **OK** > **Close**.
6. On the Reverse Proxy page, click **OK**.
7. On the Server Configuration page, click **Reverse Proxy / Authentication**.
8. Click **OK** twice to return to the Access Gateways page.

9 On the Access Gateways page, click **Update**.

10 Verify that the trusted relationship between the Identity Server and the Access Gateway has been reestablished:

10a Enter the URL to a protected resource on the Access Gateway.

10b Complete one of the following:

- If you are prompted for login credentials, enter them. The trusted relationship has been reestablished.
- If you receive a 100101043 or 100101044 error, the trusted relationship has not been established. For information about solving this problem, see "Troubleshooting 100101043 and 100101044 Liberty Metadata Load Errors" in the *NetIQ Access Manager Appliance 3.2 SP2 Identity Server Guide*.

3 Clustering and Fault Tolerance

For additional capacity and for failover, you can cluster a group of NetIQ Access Manager Appliances and configure them to act as a single server.

You can achieve load balancing and fault tolerance by installing multiple instances of Access Manager Appliance. You can cluster any number of Identity Servers, Access Gateways, SSL VPNs, and up to three of Administration Consoles. The first three nodes of Access Manager Appliance contain the Administration Console, Identity Server, Access Gateway, and SSL VPN. Fourth installation onwards, the node will have all components except the Administration Console. We recommend that you install at least one secondary Access Manager Appliance.

Clustering of NetIQ Access Manager Appliances enables the following features:

- **Fault Tolerance for Configuration Store:** You can install an Access Manager Appliance and configure it as a secondary Administration Console.
- **Automatic clustering of group of Identity Servers and group of Access Gateways:** Identity Servers and Access Gateways in each Access Manager Appliance will be grouped into respective clusters.
- **Configuration Synchronization:** You configure the cluster, and the configuration is synchronized to all members of the cluster.
- **User Session Sharing:** Each cluster member can handle sessions held by another server in the cluster. After a session is established, the same member usually handles all requests for that session. However, if that cluster member is not available to handle a request, another member steps in and processes the request.

  NOTE: This is not applicable for the Administration Console.

- Section 3.1, “Installing Secondary Versions of Access Manager Appliance,” on page 31
- Section 3.2, “Modifying Cluster Configuration,” on page 35
- Section 3.3, “Configuration Tips for the L4 Switch,” on page 39
- Section 3.4, “Using a Software Load Balancer,” on page 45

3.1 Installing Secondary Versions of Access Manager Appliance

The Administration Console contains an embedded version of eDirectory, which contains all the configuration information for the Access Manager Appliance. It also contains a server communications module, which is in constant communication with the Access Manager modules. If the Administration Console goes down and you have not installed any secondary consoles, your Access Manager components also go down and your protected resources become unavailable.

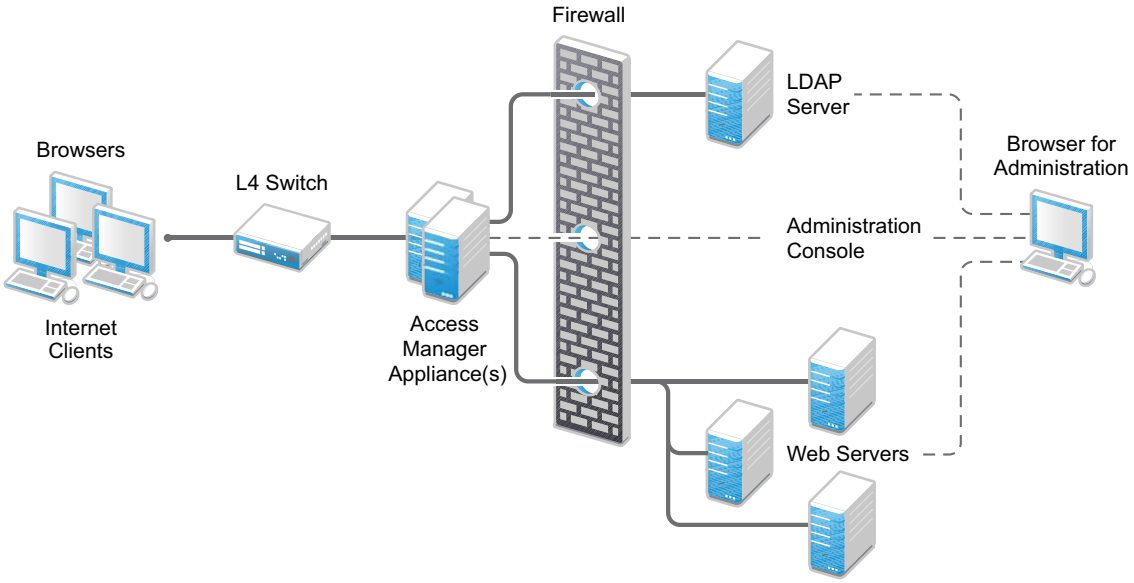
- Section 3.1.1, “Configuration Notes,” on page 32
- Section 3.1.2, “Prerequisites,” on page 33

- Section 3.1.3, “Installing a Secondary Access Manager Appliance,” on page 33
- Section 3.1.4, “Understanding How the Consoles Interact with Each Other and Access Manager Devices,” on page 34

3.1.1 Configuration Notes

- “A Note about Layer 4 Switch” on page 32
- “Services of the Real Server” on page 33
- “A Note about Service Configuration” on page 33
- “A Note about Alteon Switches” on page 33

The following figure illustrates the components and process flow that make up a basic configuration.



A Note about Layer 4 Switch

A cluster of Access Manager Appliances should reside behind a Layer 4 (L4) switch. Clients access the virtual IP address of the cluster presented on the L4 switch, and the L4 switch alleviates server load by balancing traffic across the cluster.

Whenever a user accesses the virtual IP address assigned to the L4 switch, the system routes the user to one of the Access Manager Appliances in the cluster, as traffic necessitates.

IMPORTANT: Using a DNS round robin setup instead of an L4 switch for load balancing is not recommended. The DNS solution works only as long as all members of the cluster are working and in a good state. If one of them goes down and traffic is still sent to that member, the entire cluster is compromised and all devices using the cluster start generating errors.

Services of the Real Server

A user's authentication remains on the real (authentication) server cluster member that originally handled the user's authentication. If this server malfunctions, all users whose authentication data resides on this cluster member must reauthenticate unless you have enabled session failover. For more information about this feature, see "Configuring Session Failover" in the *NetIQ Access Manager Appliance 3.2 SP2 Identity Server Guide*.

Requests that require user authentication information are processed on this server. When the system identifies a server as not being the real server, the HTTP request is forwarded to the appropriate cluster member, which processes the request and returns it to the requesting server.

A Note about Service Configuration

If your L4 switch can perform both SSL and non-SSL health checks, you should configure the L4 switch only for the services that you are using in your Access Manager configuration. For example, if you configure the SSL service and the non-SSL service on the L4 and the base URL of your Identity Server configuration is using HTTP rather than HTTPS, the health check for the SSL service fails. The L4 switch then assumes that all the Identity Servers in the cluster are down. Therefore, make sure you enable only the services that are also enabled on the Identity Server.

A Note about Alteon Switches

When you configure an Alteon switch for clustering, direct communication between real servers must be enabled. If direct access mode is not enabled when one of the real servers tries to proxy another real server, the connection fails and times out.

To enable direct communication on the Alteon:

1. Go to **cfg > slb > adv > direct**.
2. Specify *e* to enable direct access mode.

3.1.2 Prerequisites

- ❒ An L4 server is installed. The LB algorithm can be anything (hash/sticky bit), defined at the Real server level.
- ❒ Persistence (sticky) sessions enabled on the L4 server. You usually define this at the virtual server level.

NOTE: If Access Manager Appliance is configured with public and private interface, the back channel communication will use the private interface. To allow this back channel communication on private interface, modify the NAM-RP configuration to listen on private and public interfaces. For more information, see "Managing Reverse Proxies and Authentication"in *NetIQ Access Manager Appliance 3.2 SP2 Access Gateway Guide*.

3.1.3 Installing a Secondary Access Manager Appliance

1. Insert the CD containing the software.

Most of the installation process is same for a secondary appliance as for a primary. If this is a second or third appliance, the Administration Console will be configured for the fault tolerance. While installing a secondary appliance:

- Deselect the **Primary** check box.
- Enter the IP address of the primary Administration Console.
- Enter user name and password of the primary Administration Console.

Installation of the secondary appliance becomes interactive after the installation of operating system in the following cases:

- (Conditional) if this is the fourth appliance: The number of Administration Consoles in a cluster is restricted to three. If more appliances are added into the cluster, the system will ask whether you want proceed with the installation of rest of the components other than Administration Console.
- (Conditional) if time is not synchronized between the primary and secondary appliances. The system will prompt a message asking you to re-try the time synchronization or to proceed without synchronization.

If you have firewalls separating your Identity Servers or your L4 switch does not support port translation, you can use iptables to translate the port

Configure the details on the Administration Console Configuration page as specified in step 9 in the "Installing the Access Manager Appliance"section of *NetIQ Access Manager Appliance 3.2 SP2 IR1 Installation Guide*.

2 Continue with the installation process.

The Identity Provider and the Access Gateway from the secondary appliance are automatically clustered with the primary appliance. If this is second or third secondary appliance, the configuration store will be configured for the fault tolerance. It is recommend that you install at least one secondary console.

After successful installation, the appliance points to the Access Manager Appliance's IP address for the Web server, and the Identity Server points to the local user store. If a cluster is configured for Access Manager Appliance and if primary appliance is down, you cannot authenticate because the user store is on primary and they cannot access the resources because it points to the Web server on primary. Hence, it is advised to change the IP address of the Web server configured in the master proxy service to point to your test or production Web server, and change the Identity Server's configuration to point to an external user store.

3.1.4 Understanding How the Consoles Interact with Each Other and Access Manager Devices

The primary and secondary consoles use eDirectory synchronization to keep their configuration databases current.

WARNING: As long as the primary console is running, all configuration changes should be made at the primary console. If you make changes at both a primary console and a secondary console, browser caching can cause you to create an invalid configuration.

Access Manager Appliance devices use the secondary console only when the primary console is down. Therefore, if a secondary console goes down while the primary console is running, the devices are notified. But they continue to run by using the primary console for configuration information. The secondary console can be down for as long as required to fix the problem without affecting the other Access Manager Appliance devices.

When the primary console goes down, all of the devices discover this and switch to using the secondary console. This can take a few minutes, because each device has its own trigger for checking in with the Administration Console. After the device has switched to using the secondary console, it continues to run just as it did when it was communicating with the primary console. When the primary console comes back online, all of the devices discover this and switch back to using the primary console. Again, this can take a few minutes.

Not all tasks are available from the secondary console:

- "Tasks Requiring the Primary Console" on page 35
- "Tasks Available from the Secondary Console" on page 35

Tasks Requiring the Primary Console

Backup and Restore: Backup and restore must be run on the primary console. When the restore has completed, you must restart Tomcat on all secondary consoles.

Enter the following command:

```
/etc/init.d/novell-ac restart
```

For more information about backup and restore, see "Backing Up and Restoring" in the *NetIQ Access Manager Appliance 3.2 SP2 Administration Console Guide*.

Tasks Available from the Secondary Console

When the primary console goes down, the secondary console can be used for the following tasks:

- Administrators can make configuration changes on a secondary console, and these changes are sent to the Access Manager components.
- Access Manager Appliance components can use the secondary console to access their configuration information and to respond to configuration changes. As soon as the primary console comes back online, the components revert to using the primary machine, but they continue to accept commands from the secondary consoles.

3.2 Modifying Cluster Configuration

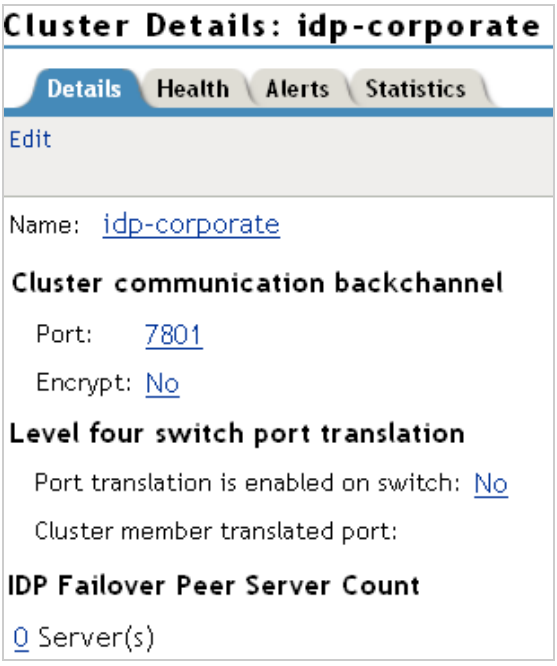
This section describes how to modify the cluster configurations:

- Section 3.2.1, "Modifying Identity Provider Cluster Configuration," on page 36
- Section 3.2.2, "Modifying Access Gateways Cluster Configuration," on page 37
- Section 3.2.3, "Modifying SSL VPN Server Cluster Configuration," on page 38

3.2.1 Modifying Identity Provider Cluster Configuration

1 In the Administration Console, click **Devices > Identity Servers**, then click the configuration name you created for the cluster.

2 On the Cluster Details page, click the configuration name.



3 Fill in the following fields as required:

Name: Lets you change the name of the Identity Server cluster configuration.

Cluster Communication Backchannel: Provides a communications channel over which the cluster members maintain the integrity of the cluster. For example, this TCP channel is used to detect new cluster members as they join the cluster, and to detect members that leave the cluster. A small percentage of this TCP traffic is used to help cluster members determine which cluster member would best handle a given request. This back channel should not be confused with the IP address/port over which cluster members provide proxy requests to peer cluster members.

- **Port:** Specifies the TCP port of the cluster back channel on all of the Identity Servers in the cluster. 7901 is the default TCP port.

  Because the cluster back channel uses TCP, you can use cluster members on different networks. However, firewalls must allow the port specified here to pass through. To do so, use the port number plus 1 for additional devices in the cluster. For example, if you use four devices, your port numbers would be 7901, 7902, 7903, and 7904.

- **Encrypt:** Encrypts the content of the messages that are sent between cluster members.

NOTE: The Level Four Switch Port Translation feature is not required for Access Manager Appliance as Identity Server cluster is accelerated through Access Gateway.

IDP Failover Peer Server Count: Enables session failover. For more information about this feature, see "Configuring Session Failover" in the *NetIQ Access Manager Appliance 3.2 SP2 Identity Server Guide*.

4 Click **OK**.

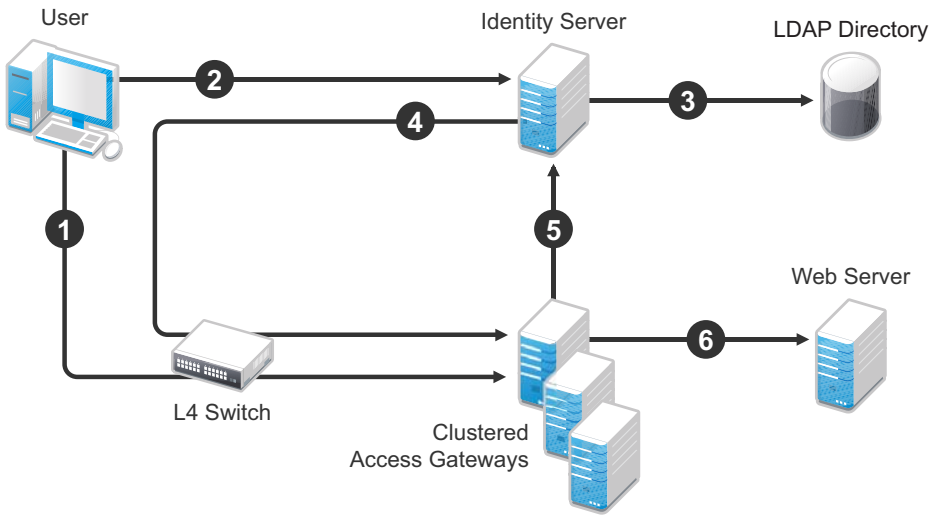
5 Under **Cluster Members**, you can refresh, start, stop, and update health from the server.

3.2.2 Modifying Access Gateways Cluster Configuration

A cluster of the Access Gateways must reside behind a Layer 4 (L4) switch. Clients access the virtual IP on the L4, and the L4 alleviates server load by balancing traffic across the cluster of Access Gateways. Whenever a user enters the URL for an Access Gateway resource, the request is routed to the L4 switch, and the switch routes the user to one of the Access Gateways in the cluster, as traffic necessitates.

Figure 3-1 illustrates the flow of a user request when the Access Gateways are clustered behind an L4 switch.

Figure 3-1 *Clustering Access Gateways*



1. The user requests access to a protected resource by sending a request to the L4 switch. The request is sent to one of the Access Gateway servers in the cluster.
2. The Access Gateway redirects the request to the Identity Server for authentication. The Identity Server presents the user with a login page, requesting a user name and a password.
3. The Identity Server verifies the user's credentials with the directory.
4. The validated credentials are sent through the L4 switch to the same Access Gateway that first received the request.
5. The Access Gateway verifies the user credentials with the Identity Server.
6. If the credentials are valid, the Access Gateway forwards the request to the Web server.

If the Access Gateway where the user's session was established goes down, the user's request is sent to another Access Gateway in the cluster. This Access Gateway pulls the user's session information from the Identity Server. This allows the user to continue accessing resources, without having to reauthenticate.

IMPORTANT: Using a DNS round robin setup instead of an L4 switch for load balancing is not recommended. The DNS solution works only as long as all members of the cluster are working and in a good state. If one of them goes down and traffic is still sent to that member, the entire cluster is compromised and starts generating errors.

The following sections describe how to set up and manage a cluster of Access Gateways.

Prerequisites

- ❒ An L4 switch installed. You can use the same switch for an Identity Server cluster and an Access Gateway cluster, provided that you use different virtual IPs.
- ❒ One or more Access Gateways installed.

  When you install each new Access Gateway, configure it to use the same Administration Console.
- ❒ Your DNS server must to be configured to resolve the published DNS names that you specify for your proxy services to the L4 switch.
- ❒ Enabling persistent (sticky) sessions on the L4 switch is highly recommended, but not required.

Configuring a Cluster

Complete the following steps:

1. In the Administration Console, click **Access Gateways** > Edit AG-Cluster.
2. To configure the cluster, click **Access Gateways** > **Edit**.

   A cluster of Access Gateways has the same configuration options as a single Access Gateway. The only difference is that for some options you need to select the Access Gateway to configure. For example, the **Date & Time** option allows you to set the time separately for each member of the cluster.

   Applying the configuration to a cluster is slightly different. You have the option to apply the changes to all servers in the cluster by selecting the **Update All** option, or to apply them to one server at a time by selecting the **Update** option for each server. When you update the servers one at time, your site remains up. For more information about the **Update** and **Update All** options, see "Configuration Options" in the *NetIQ Access Manager Appliance 3.2 SP2 Access Gateway Guide*.

   If you prefer to apply changes to the servers one at time, you should save the changes to the configuration datastore. To do this, click **OK** on the Server Configuration page. (The **OK** buttons on the other configuration pages save the changes to browser cache.) If your session times out before you update all servers in the cluster and the changes have been saved only in browser cache, the changes are lost and are not applied to the servers that are still in an **Update** status.
3. (Conditional) If the Access Gateways in the cluster have multiple network adapters or IP addresses, you need to configure the listening address for each reverse proxy.

   If this is not the address where you want the reverse proxy to listen for requests, click Access Gateways > Edit > [Name of Reverse Proxy], select the Access Gateway as the Cluster Member, then enable the Listening Address you want to use.

3.2.3 Modifying SSL VPN Server Cluster Configuration

You can cluster the high-bandwidth SSL VPN servers to provide load balancing and fault tolerance capabilities and act as a single server.

For more information about configuring the SSL VPN cluster by using the Access Gateway, see "Clustering SSL VPN by Using an L4 Switch" in the *NetIQ Access Manager Appliance 3.2 SP2 SSL VPN Server Guide*.

3.3 Configuration Tips for the L4 Switch

You need to configure the DNS server to resolve the Base URL of the Identity Server to the Identity Server VIP on the L4 switch. In the Access Manager Appliance, the Base URL of the Identity Server is same as the Base URL of the Access Gateway.

The Identity Server is accessible only using the Base URL of the Access Gateway and cannot be accessed using a public IP address. So, to configure the Identity Server cluster, the Access Gateway cluster configuration should dedicate its base reverse proxy service only for the Identity Server. All other applications have be configured as separate proxy services with a separate listener.

For example: If the URL of the company is www.acme.com, the Identity Server Base URL is www.acme.com by default. Once installed the DNS name cannot be changed.

In addition to this basic setup, consider the following:

3.3.1 Sticky Bit

Each L4 switch has a slightly different method and terminology for the sticky bit or persistence bind. This bit allows a client that has established a session to be directed to the same Identity Server or Access Gateway for all requests sent during the session. This minimizes the need to forward session information between Access Gateways or between Identity Servers and thus maximizes performance.

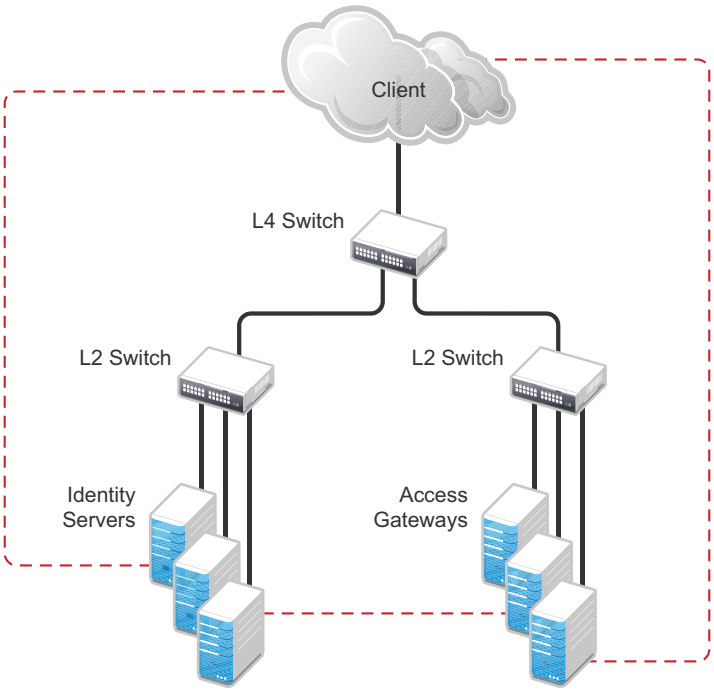
3.3.2 Network Configuration Requirements

When you set up the L4 switch, be aware of the following configuration requirements that are required to route all Access Manager traffic through the L4 switch:

Switches: When you install an L4 switch, you can plug in the machines directly to the L4 switch. For Access Manager Appliance, inner switch is not required.

Network Routing Requirements: You need to analyze your routing configuration. The Identity Servers and the Access Gateways must be connected to separate ports in the L4 switch. If there is a connection in your network that allows an Identity Server or an Access Gateway to communicate directly with a client without going through the L4 switch, the Access Gateway and the Identity Server try to establish direct communication with the client because networking protocols are configured to select the most direct route. Such a configuration causes communication problems because all traffic must be routed through the L4 switch. Figure 3-2 illustrates this problem.

Figure 3-2 *Network Configuration with a Potential Communication Problem*



If your network allows for this type of communication, you need to block the communication channels illustrated with the dotted lines.

Figure 3-2 shows each cluster type with its own L2 switch. An Access Gateway cluster and an Identity Server cluster cannot share the same L2 switch because they can see the MAC address for each other. Networking protocols are configured to use the most direct route for the communication, and the MAC address is more direct than going up to the L4 switch and back down. Such a configuration causes communication problems because all traffic between the clusters needs to be routed through the L4 switch. Using a separate L2 switch for each cluster type prevents them from gaining access to the MAC address and forces communication to take place through the L4 switch.

3.3.3 Health Checks

L4 switches use health checks to determine which cluster members are ready to receive requests and which cluster members are unhealthy and should not receive requests. You need to configure the L4 switch to monitor the heartbeat URL of the Identity Servers and Access Gateways, so that the L4 switch can use this information to accurately update the health status of each cluster member.

The procedure is slightly different for the Identity Servers and Access Gateways:

- "Health Checks for the Identity Server" on page 40
- "Health Checks for the Access Gateway" on page 41

Health Checks for the Identity Server

The Administration Console uses the heartbeat URL to display the health status of the Identity Servers. The Identity Server heartbeat is the DNS name of the Identity Server plus the following path:

```
/nidp/app/heartbeat
```

L4 switches require you to use IP address rather than the DNS name. If the IP address of the Identity Server is 10.10.16.50, and you have configured the Identity Server for HTTPS, the heartbeat has the following URL:

```
https://10.10.16.50:8443/nidp/app/heartbeat
```

You need to configure the L4 switch to use this heartbeat to perform a health check. If you have configured SSL on the Identity Servers and your L4 switch has the ability to do an SSL L7 health check, you can use HTTPS. The SSL L7 health check returns a value of `200 OK`, indicating that everything is healthy; any other status code indicates an unhealthy state.

For a Foundry switch, the L7 health check script string should look similar to the following when the hostname is nidp1 and the IP address is 10.10.16.50:

```
healthck nidp1ssl tcp
  dest-ip 10.10.16.50
  port ssl
  protocol ssl
  protocol ssl url "GET /nipd/app/heartbeat HTTP/1.1\r\nHost: st160.lab.tst"
  protocol ssl status-code 200 200
  l7-check
```

If your switch does not support an SSL L7 health check, the HTTPS URL returns an error, usually a 404 error. Because the Identity Server heartbeat URL listens on both HTTPS and HTTP, you can use an HTTP URL for switches that do not support the SSL L7 health check. For example:

```
http://10.10.16.50:8080/nidp/app/heartbeat
```

An Alteon switch does not support the L7 health check, so the string for the health check should look similar to the following:

```
open 8080,tcp
send GET /nidp/app/heartbeat HTTP/1.1\r\nHOST:heartbeat.lab.tst \r\n\r\n
expect HTTP/1.1 200
close
```

Health Checks for the Access Gateway

External communication to the Access Gateway is typically configured to use HTTPS. In an HTTPS configuration, an L4 switch performs health checks of the Access Gateways with the published DNS name of the Access Gateway plus the following path:

```
/nesp/app/heartbeat
```

L4 switches require you to use IP address rather than the DNS name. If the IP address of the Access Gateway is 10.10.16.172, and you have configured the Access Gateway for HTTPS, the heartbeat has the following URL:

```
https://10.10.16.172:443/nesp/app/heartbeat
```

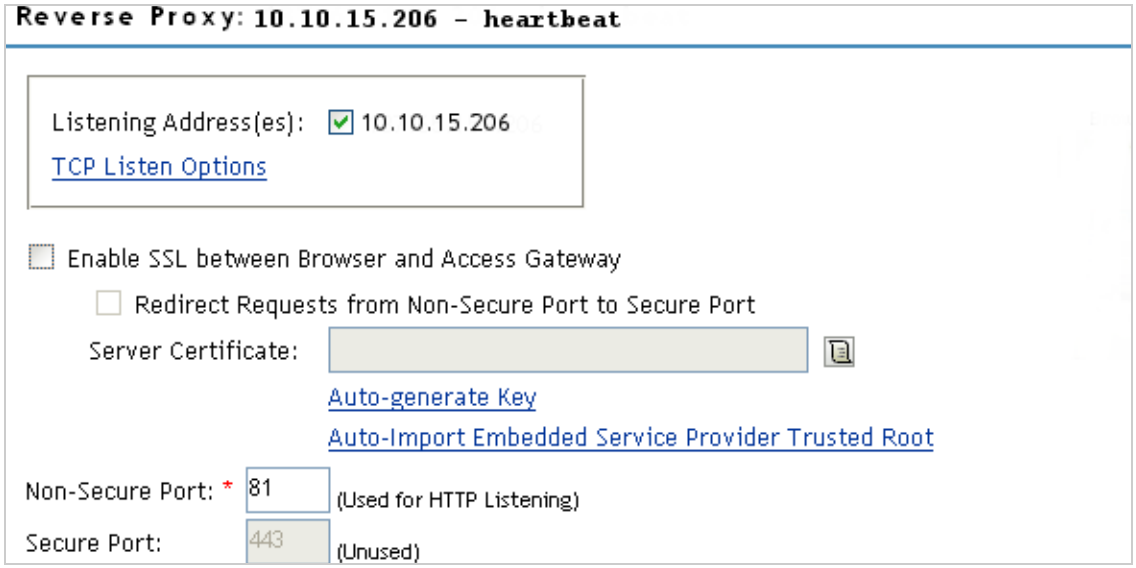
For an L4 switch to support an HTTPS query for the health of the Access Gateway, the switch must support an L7 health check. For a Foundry switch, the L7 health check script string should look similar to the following when the hostname is ag1 and the IP address is 10.10.172.

```
healthck ag1ssl tcp
  dest-ip 10.10.16.172
  port ssl
  protocol ssl
  protocol ssl url "GET /nesp/app/heartbeat HTTP/1.1\r\nHost: st160.lab.tst"
  protocol ssl status-code 200 200
  l7-check
```

If your L4 switch does not support an SSL L7 health check, the HTTPS health check URL returns an error, usually a 404 error. To solve this problem, you can create a specialized reverse proxy that opens a non-SSL port for the heartbeat URL. The following instructions configure this reverse proxy to use port 81, because port 80 on the specified IP address is reserved for redirects to the SSL port.

To create a reverse proxy for the health check:

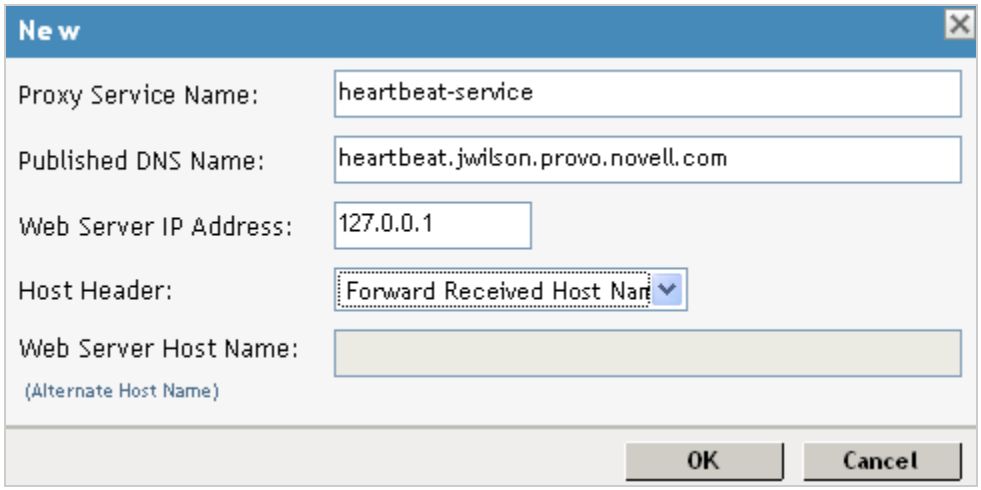
1. In the Administration Console, click **Access Manager** > **Access Gateways** > **Edit** > **Reverse Proxy / Authentication**.
2. To create an additional reverse proxy service (such as *heartbeat*), click **New**, then specify a name.



3. Change the **Non-Secure Port** to 81.

   You configure the Access Gateway to listen on the same IP address as the service using port 443. For non-SSL, port 81 is recommended. Do not use port 80.

   For proper heartbeat information when there are multiple IP addresses configured in your Access Gateway, ensure that you configure the reverse proxy service created for the heartbeat URL to listen in the same IP address as the authenticating reverse proxy service.
4. Click **New** to create the proxy service.



5. Configure the following fields:

Proxy Service Name: Specify a name that identifies the purpose of this proxy service.

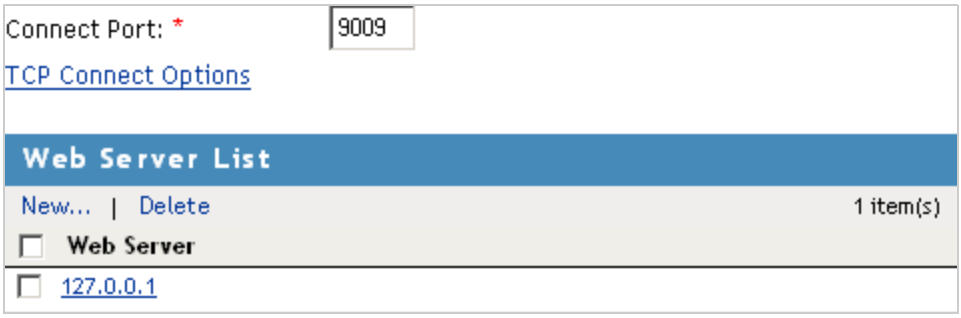
Published DNS Name: Specify a second DNS name that resolves to the VIP of the Access Gateways on the L4 switch. For example, if the DNS name is jwilson.provo.novell.com for the Access Gateways, you could use heartbeat.jwilson.provo.novell.com for the second name.

Web Server IP Address: Specify the internal address:127.0.0.1.

Host Header: Select **Forward Received Host Name**. This field is not used.

6 Click **OK**.

7 On the Reverse Proxy page, click the new proxy service, then click **Web Servers**.



8 Change the **Connect Port** value on the Web Servers page to 9009.

The service provider (ESP) in the Access Gateway that provides the heartbeat service listens on 127.0.0.1:9009.

9 Click **Protected Resources**.

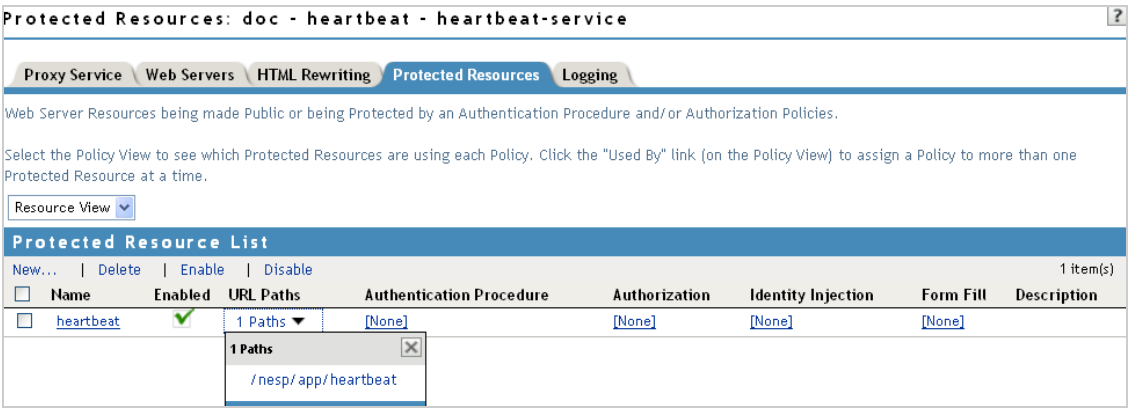
10 Click **New**, then specify a name.

11 In the URL Path List, click /*, and modify the path to contain the following value:

```
/nesp/app/heartbeat
```

This is the path to the heartbeat application.

12 Click **OK** twice. Your protected resource for the heartbeat application should look similar to the example below.



The heartbeat of this Access Gateway is available from the following URL (See Step 4.):

```
http://heartbeat.jwilson.provo.novell.com:81/nesp/app/heartbeat
```

If the protected resource is configured with a path of / or /*, the solution works but it can be vulnerable to attacks because the configuration opens the ESP over a non-SSL port. Restricting the resource to `/nesp/app/heartbeat` automatically denies access to the ESP except for the heartbeat.

13 Click **OK** and apply the changes to the configuration.

14 Add a line similar to the health check script:

For a Foundry switch, your string should look similar to the following if the hostname is ag1 and the IP address is 10.10.16.172:

```
healthck ag1 tcp
  dest-ip 10.10.16.172
  port http
  protocol http
  protocol http url "GET /nesp/app/heartbeat HTTP/1.1\r\nHost:st160.lab.tst"
  protocol http status-code 200 200
  l7-check
```

For an Alteon switch, your string should look similar to the following if the hostname is ag1 and the IP address is 10.10.16.172:

```
open 81,tcp
send GET /nesp/app/heartbeat HTTP/1.1\r\nHOST:heartbeat.lab. tst\r\n\r\n
expect HTTP/1.1 200
close
```

3.3.4 Real Server Settings Example

After setting up the health checks, you need to configure the real server settings. The following is an example from a Foundry switch.

```
Current real servers settings:
  1: 149.44.171.116, enabled, name 152, weight 1, timeout 10 mins, maxcon 200000
    backup none, inter 2, retry 4, restr 8
    remote disabled, proxy enabled, submac disabled
    cookie assignment server: disabled
    exclusionary string matching: disabled
    service ports: 8443 8080
    real ports:
      8443: vport 8443, group 1, pbind clientip
          virtual server:  1, 149.44.174.220,  enabled
      8080: vport 8080, group 1, pbind clientip
          virtual server:  1, 149.44.174.220,  enabled
  2: 149.44.174.51, enabled, name brie, weight 1, timeout 10 mins, maxcon 200000
    backup none, inter 2, retry 4, restr 8
    remote disabled, proxy enabled, submac disabled
    cookie assignment server: disabled
    exclusionary string matching: disabled
    service ports: 8443 8080
    real ports:
      8443: vport 8443, group 1, pbind clientip
          virtual server:  1, 149.44.174.220,  enabled
      8080: vport 8080, group 1, pbind clientip
          virtual server:  1, 149.44.174.220,  enabled
```

3.3.5 Virtual Server Settings Example

After setting up the real server settings, you need to configure the virtual server settings. The following is an example from a Foundry switch.

```
Current virtual servers settings:
  1: 149.44.174.220, enabled, dname idp
    virtual ports:
      8443: rport 8443, group 1, pbind clientip, frags
          real servers:
             1: 149.44.171.116,  weight 1,  enabled, backup none
             2: 149.44.174.51,   weight 1,  enabled, backup none
      8080: rport 8080, group 1, pbind clientip, frags
          real servers:
             1: 149.44.171.116,  weight 1,  enabled, backup none
             2: 149.44.174.51,   weight 1,  enabled, backup none
```

3.4 Using a Software Load Balancer

Instead of using an L4 switch, you can cluster the Identity Servers and the Access Gateways behind a software load balancer that runs in Layer 7. Each manufacturer uses slightly different terminology, but the basic steps are quite similar. You need to create the following types of objects:

- Pools to specify how load balancing occurs, such as round robin.
- Persistence classes to be used within the pools to enable the sticky bit or to keep state so that a connection is sent to the same device.
- Monitors to be used within the pools for monitoring the health heartbeat of the device.
- Virtual servers to set up the ports and protocols for the pools.
- Traffic IP groups where the virtual IP addresses are set up and tied to the virtual servers.

Because the software actually runs in Layer 7, it does not require any special networking setup and it runs on standard server hardware.

As an example, the following instructions explain how to configure the Zeus ZXTM Load Balancer with HTTP and HTTPS for the Identity Server and Access Gateway. For more information about this product, see Zeus Technology (http://www.zeus.com/).

1. Create a persistence class for HTTPS.

   ```
   HTTPS > SSL Session ID
   ```

2. Create four monitors, two for the Identity Servers and two for the Access Gateways.

   2a Use the following path to specify a path for HTTP:

   Access Manager Appliance: `/nesp/app/heartbeat`

   2b Configure the following parameter for the monitors:

   HTTP: timeout=10 seconds, use_ssl=no, host_header: <domain>, body_regex: Success

   Replace <*domain*> with the DNS name of the Access Manager device

3. Create four pools, one for each monitor. Configure each pool with the following parameters:

   ```
   Load _balancing: Round Robin
   persistence: <new class created>
   max_reply_time: 10
   ```

For an HTTP resource, replace <*new class created*> with the HTTP class you created.

4 Create four virtual servers, one for each port. Configure each with the following parameters:

```
Protocol: <scheme>
Port: <port>
Pool: <pool created>
```

Replace <*scheme*> with HTTP or HTTPS.

Replace <*port*> with one of the following values: 80,8080,443, or 8443.

Replace <*pool created*> with one of the pools you created in Step 3.

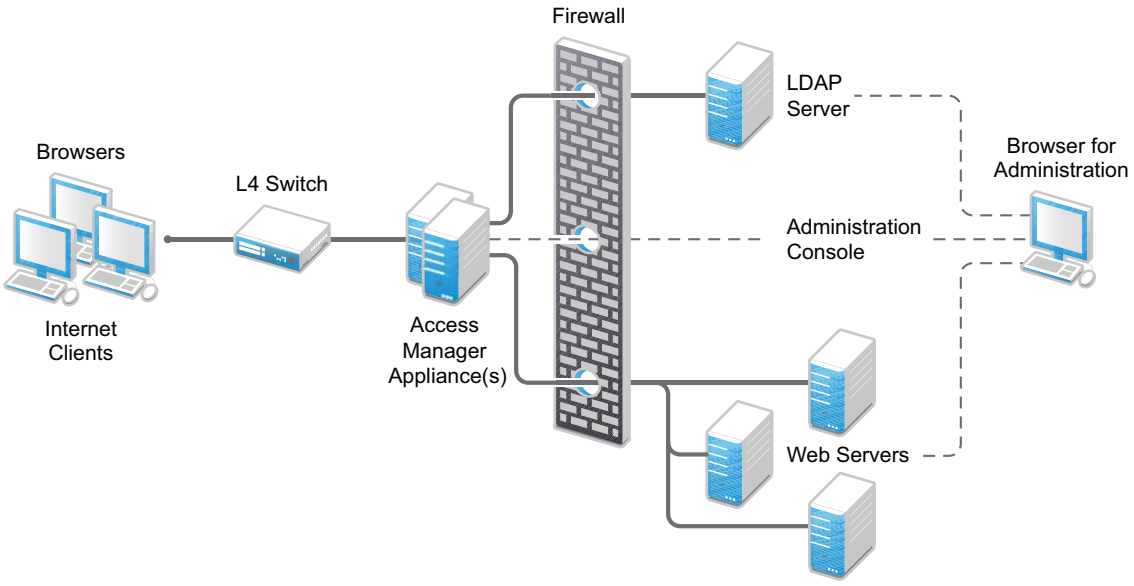
5 Create two traffic manager groups, one for the Identity Servers and one for the Access Gateway. This is where the virtual IP address is set up.

6 Start the traffic groups.

4 Setting Up Firewalls

Access Manager Appliance is not a firewall; it should be used with firewalls. Figure 4-1 illustrates a simple firewall setup for a basic Access Manager Appliance configuration .

Figure 4-1 *Access Manager Appliance and Firewall*



The first firewall separates the Access Manager Appliance from the Internet, allowing browsers to access the resources through specific ports. This is one of many configurations possible. This section describes the following:

- Section 4.1, “Required Ports,” on page 47
- Section 4.2, “Sample Configurations,” on page 49

4.1 Required Ports

The following table lists the ports that need to be opened when a firewall separates Access Manager Appliance from Internet.

With these tables, you should be able to place Access Manager Appliance of your system anywhere within your existing firewalls and know which ports need to be opened in the firewall.

Component	Port	Description
NTP Server	UDP 123	Access Manager components must have time synchronized or authentication fails. We highly recommend that all components be configured to use an NTP (network time protocol) server. Depending upon where your NTP server is located in relationship to your firewalls, you might need to open UDP 123 so that the Access Manager component can use the NTP server.
DNS Servers	UDP 53	Access Manager components must be able to resolve DNS names. Depending upon where your DNS servers are located, you might need to open UDP 53 so that the Access Manager component can resolve DNS names.
Remote Linux Administration Workstation	TCP 22	If you use SSH for remote administration and want to use it for remote administration of Access Manager components, you need to open TCP 22 to allow communication from your remote administration workstation to your Access Manager components.
Access Manager Appliance	TCP 1443	For communication from the Administration Console to the devices.
	TCP 8444	For communication from the devices to the Administration Console.
	TCP 1289	For communication from the devices to the Audit server on the Administration Console.
	TCP 524	For NCP certificate management with NPKI. The port needs to be opened so that both the device and the Administration Console can use the port.
	TCP 636	For secure LDAP communication from the devices to the Administration Console.
	TCP 524	Required to synchronize the configuration data store.
	TCP 636	Required for secure LDAP communication.
	TCP 8080, 8443	Used for Tomcat communication.
LDAP User Store	TCP 524	Required only if the user store is eDirectory. When configuring a new eDirectory user store, NCP is used to enable Novell SecretStore by adding a SAML authentication method and storing a public key for the Administration Console. It is not used in day-to-day operations.

Component	Port	Description
Browsers	TCP 8080	For HTTP communication from browsers to the Administration Console.
	TCP 8443, 2443, 2080.	For HTTPS communication from browsers to the Administration Console.
	TCP 8028, 8030	To use iMonitor or DSTrace from a client to view information about the configuration store on the Administration Console.
	TCP 80	For HTTP communication from the client to the Access Gateway. This is configurable.
	TCP 443	For HTTPS communication from the client to the Access Gateway. This is configurable.
Web Servers	TCP 80	For HTTP communication from the Access Gateway to the Web servers. This is configurable.
	TCP 443	For HTTPS communication from the Access Gateway to the Web servers. This is configurable.
OpenVPN	UDP 7777	For OpenVPN server communication. This is the default port for access to the SSL VPN, but it can be configured to use UDP 443.
Application Servers (E-mail, Telnet, Thin Client)	TCP 22	For SSH communication from the SSL VPN to the application server.
	TCP 23	For Telnet communication from the SSL VPN to the application server.
	Application ports	Specific to the application that SSL VPN is providing access to.
Firewall on same machine as the SSL VPN	tun0	SSL VPN creates a tunnel that needs to be open on the internal networks list of the machine. For configuration information, see the following Note.

NOTE: On SLES 11, you can edit this file or use YaST to configure UDP ports and internal networks.

4.2 Sample Configurations

4.2.1 Access Manager Appliance in DMZ

First Firewall

If you place a firewall between the browsers and Access Manager Appliance, you need to open ports so that the browsers can communicate with the Access Gateway and the Identity Server and the Identity Server can communicate with other Identity Providers.

See, Figure 4-1 on page 47

Table 4-1 *Ports to Open in the First Firewall*

Port	Purpose
TCP 80	For HTTP communication.
TCP 443	For HTTPS communication.
Any TCP port assigned to a reverse proxy or tunnel.	
TCP 8080	For HTTP communication with the Identity Server.
TCP 8443	For HTTPS communication with the Identity Server.
TCP 8445	For HTTP Identity Provider introductions. If you do not enable Identity Provider introductions, you do not need to open this port. .
TCP 8446	For HTTPS Identity Provider introductions. If you do not enable Identity Provider introductions, you do not need to open this port.

The SSL VPN server needs the following port opened on the first firewall if clients are accessing the SSL VPN server directly:

Table 4-2 *Ports to Open in the First Firewall for SSL VPN*

Port	Purpose
TCP 7777	For client communication. This is the default port, but it can be configured to use TCP 443.

Second Firewall

The second firewall separates the Web servers, LDAP servers, and the Administration Console from the Identity Server and the Access Gateway. You need the following ports opened in the second firewall:

Table 4-3 *Ports to Open in the Second Firewall*

Port	Purpose
TCP 80	For HTTP communication with Web servers.
TCP 443	For HTTPS communication with Web servers.
Any TCP connect port assigned to a Web server or to a tunnel.	
TCP 1443	For communication from the Administration Console to the devices.
TCP 8444	For communication from the devices to the Administration Console.
TCP 1289	For communication from the devices to the Novell Audit server installed on the Administration Console. If you do not enable auditing, you do not need to open this port.
TCP 524	For NCP certificate management in NPKI. The port needs to be opened so that both the device and the Administration Console can use the port.
TCP 636	For secure LDAP communication of configuration information.

You need to open ports on the second firewall according to the offered services.

Table 4-4 *Ports to Open in the Second Firewall for SSL VPN*

Port	Purpose
TCP 22	For SSH
TCP 23	For Telnet

5 Setting Up Federation

Federation allows a user to associate two accounts with each other. This allows the user to log into one account and access the resources of the other account without logging in to the second account. It is one method for providing single sign-on when a user has accounts in multiple user stores.

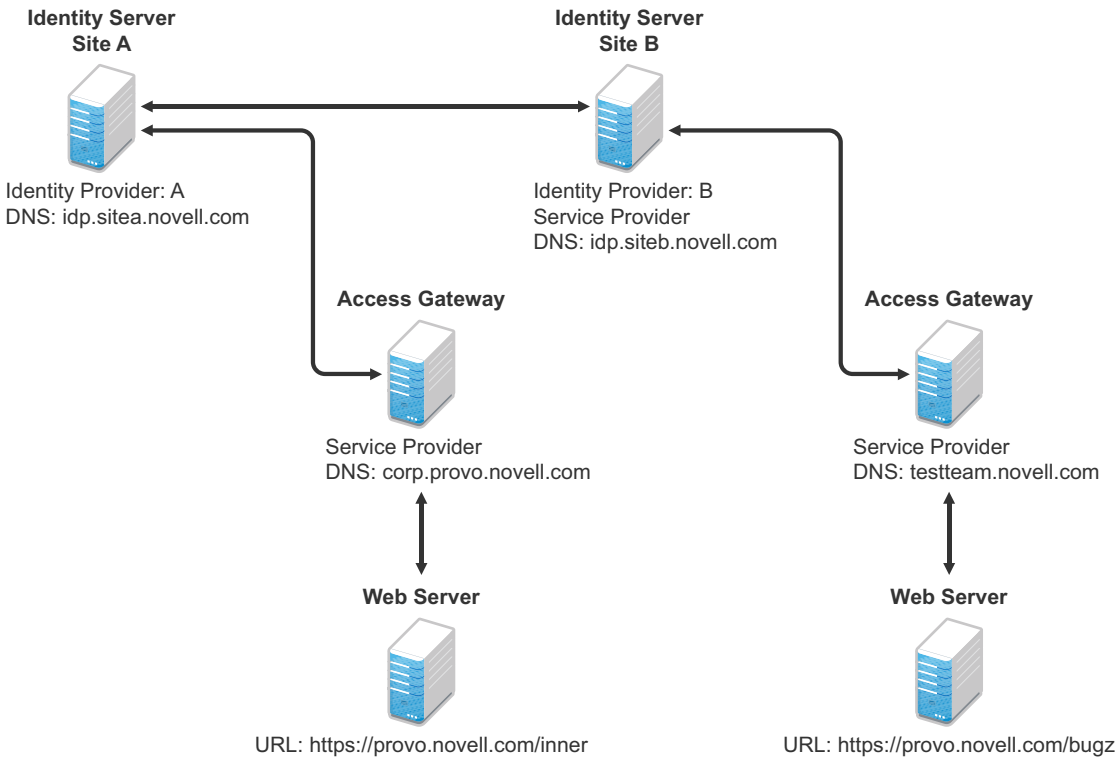
This section includes:

5.1 Understanding a Simple Federation Scenario

Suppose Company A has a centralized user store that does the authentication for most of the company’s internal resources on its inner Web site. But Company A also has a customer feedback application that employees and customers need access to, and for this application, a second user store has been created. This user store contains both employee and customer user accounts. The centralized user store cant be used, because it can contain only employee accounts. This means that the employee must log in to both accounts to access both the inner Web site and the customer feedback application. With federation, the employee can access the resources of both sites by using a single login.

Figure 5-1 illustrates such a network configuration where the user accounts of Site A are configured to federate with the user accounts at Site B.

Figure 5-1 *Using Federated Identities*



In this configuration, Site A is the Identity Server for the corporate resources, and the employees authenticate to this site and have access to the resources on the Web server with the URL of `https://provo.novell.com/inner`. Site B is the Identity Server for the Bugzilla application, and both employees and customers authenticate to this site to have access to the resources of the Web server with the URL of `https://provo.novell.com/bugz`. After an account has been federated, the user can log in to Site A and have access to the resources on the Web servers of both Site A and Site B.

In this scenario, Site B is not as secure a site as Site A, so federation is configured to go only one way, from Site A to Site B. This means that users who log in to Site A have access to the resources at Site A and B, but users who log in to Site B have access only to the resources at Site B. Federation can be configured to go both ways, so that it doesn't matter whether the user logs into Site A or Site B. When federation is configured to be bidirectional, both sites need to be equally secure.

The Access Gateways in Figure 5-1 are service providers and are configured to use the Identity Servers as identity providers. The trusted relationship is automatically set up for you when you specify authentication settings for the Access Gateway and select an Identity Server Cluster.

Federation can be set up between providers in the same company or between providers of separate companies. For example, most companies have contracts with other companies for their user's health benefits and retirement accounts. Their users have accounts with these companies. These accounts can be federated with the user's employee account when both companies agree to set up the trusted relationship.

5.2 Configuring Federation

Federation requires the configuration of a trusted relationship between an identity provider and a service provider. Figure 5-2 illustrates setting up federation between two identity servers, because a NetIQ Identity Server can act as either an identity provider or a service provider.

Figure 5-2 *Configuring Trust Between Site A and Site B*

Site A must be configured to trust Site B as a service provider, and Site B must be configured to trust Site A as an identity provider. Until this two-way trust is established, federation cannot occur.

Before setting up a trusted relationship, you must make the following decisions:

Protocol: The Identity Server supports SAML 1.1, SAML 2.0, and Liberty. You need to decide which of these protocols to use. If no user interaction is needed, SAML 1.1 is probably a good choice. The SAML 2.0 and Liberty protocols permit user interaction when federating. The user decides whether to federate (link) the accounts and must be logged in at both sites to accomplish this. Liberty offers an additional service, not available with SAML 2.0, that allows the user to select attributes that can be shared with the service provider.

The instructions in this documentation, starting in Section 5.2.1, “Prerequisites,” on page 56, use the Liberty protocol. They also indicate how to configure for the SAML 2.0 and SAML 1.1 protocols.

Trust Relationship: You need to decide whether the trusted relationship is going to be from Site A to Site B, from Site B to Site A, or bidirectionally from Site A to Site B and from Site B to Site A. Federation is set up to go from the most secure site to the less secure site. The only time federation is set up to be bidirectional is when both sites are equally secure. The scenario described in Figure 5-1 on page 54 is an example of a trusted relationship that you would want to go only one way, from Site A to Site B, because Site B is not as secure as Site A.

The instructions, starting in Section 5.2.1, “Prerequisites,” on page 56, explain how to set up the trusted relationship between Site A and Site B. You can easily modify them to set up the bidirectional trust relationships by substituting Site B for Site A (and vice versa) in the instructions and then repeating them for Site B

Attributes to Share: You need to decide whether there are user attributes or roles at Site A that you want to share with Site B. The attributes from Site A can be used to identify the users at Site B. Other attributes might be needed to access protected resources, for example, to satisfy the requirements of an Identity Injection policy.

For all the protocols, Section 5.3, “Sharing Roles,” on page 67 explains how to share the roles at Site A with Site B. For the SAML 1.1 protocol, the instructions starting in Section 5.2.1, “Prerequisites,” on page 56 use the LDAP mail attribute to share the user’s e-mail address.

User Identification: You need to decide how assertions can be used to map users from Site A to users at Site B. The Identity Server supports four methods:

- **Temporary:** This method allows the user access to Site B solely from the credentials of Site A. No effort is made to map the user to a user account at Site B. A temporary account is set up for the user on Site B, and when the user logs out, the account is destroyed.
- **Login:** This method requires that the user have login credentials at both Site A and Site B, and when logged in at both sites, the user can select to federate the accounts.
- **Mapped Attributes:** This method requires that the sites share attributes and that these attributes are used to create a matching expression that determines whether the user accounts match. For an added security check, the first time the accounts are matched, the user is asked to verify the match by supplying the password for Site B.

  If the match fails, you can allow the federation to fail or you can configure the method to allow the user to use the Login method or the Provisioning method.
- **Provisioning:** This method allows the user to create a new, permanent account at Site B.

The configuration instructions, starting in Section 5.2.1, "Prerequisites," on page 56, use the Login method for the SAML 2.0 and Liberty protocols and Mapped Attributes method for the SAML 1.1 protocol.

The instruction for setting up a trusted relationship between two NetIQ Identity Servers have been divided as follows:

5.2.1 Prerequisites

- ❒ A basic Access Manager Appliance configuration with the Identity Server and Access Gateway configured for SSL.

  This can be the one you set up using the instructions in either Chapter 1, "Setting Up a Basic Access Manager Appliance Configuration," on page 9 or Digital Airlines Example (https://wwwtest.netiq.com/documentation/netiqaccessmanager32/basicconfig/data/bayxa4y.html). For SSL configuration, see Chapter 2, "Enabling SSL Communication," on page 21.

  The Identity Server from this configuration becomes Site B in Figure 5-2.
- ❒ A second Identity Server with a basic configuration, an LDAP user store, and SSL. This Identity Server is configured to be Site A in Figure 5-2.
- ❒ Time synchronization must be set up for all the machines, or authentication can fail if assertions expire before they can be used.
- ❒ A DNS server must be configured to resolve the DNS names of Site A, Site B, and the Access Gateways.
- ❒ (Recommended) Logging has been enabled on the Identity Servers of Site A and Site B. See "Enabling Component Logging" in the *NetIQ Access Manager Appliance 3.2 SP2 Identity Server Guide*. Make sure that you enable at least application and protocol (Liberty, SAML1, or SAML2) logging at an Info level or higher.

5.2.2 Establishing Trust between Providers

To set up this very basic example of federation, complete the following tasks.

- "Configuring Site A to Trust Site B as a Service Provider" on page 57
- "Configuring Site B to Trust Site A as an Identity Provider" on page 58
- "Verifying the Trust Relationship" on page 60
- "Configuring User Authentication" on page 61

Configuring Site A to Trust Site B as a Service Provider

To establish trust between Site A and Site B, you must perform two tasks:

- The providers must trust the certificates of each other so you need to import the trusted root certificate of Site B to Site A.
- You must also import the metadata of Site B to Site A. The metadata allows Site A to verify that Site B is truly Site B when Site B sends a request to Site A.

The following instructions explain how to import the certificate and the metadata:

1. Log in to the Administration Console for Site A.

   The configuration for Site A can be created in the same Administration Console as Site B; it cannot be configured to be a cluster member of Site B.

2. Import the trusted root certificate of Site B into the NIDP trust store of Site A:

   2a Click **Devices** > **Identity Servers** > **Edit** > **Security** > **NIDP Trust Store**.

   2b In the Trusted Roots section, click **Auto-Import From Server**, then fill the following fields:

   Server IP/DNS: Specify the IP address or DNS name of Site B. For Site B in Figure 5-2 specify the following:

   ```
   idp.siteb.novell.com
   ```

   Server Port: Specify 8443.

   2c Click **OK**, then specify an alias for the certificate (for example, SiteB).

   You will get two certificate options: Root CA Certificate and

   Server certificate. We recommend you to select Root CA Certificate.

   2d Examine the trusted root that is selected for you.

   If the trusted root is part of a chain, make sure you select the parent and all intermediate trusted roots.

   2e Click **OK**.

   The trusted root certificate of Site B is added to the NIDP trust store.

   2f Click **Close**.

   2g Click **Devices** > **Identity Servers,** then update the Identity Server.

   Wait for the health status to return to green.

3. Configure a service provider for Site A:

   3a Click **Identity Servers** > **Edit** > **Liberty** [or **SAML 2.0** or **SAML 1.1**].

   3b Click **New**, select **Service Provider**, then fill the following fields:

   Name: Specify a name for the provider. If you plan on configuring more than one protocol, include the protocol as part of the name, such as, SiteB_Liberty

Metadata URL: Specify the URL of the Liberty metadata on Site B. For Site B in Figure 5-2, specify the following:

```
http://idp.siteb.novell.com:8080/nidp/idff/metadata
```

This example uses port 8080 to avoid any potential certificate problems that occur when the Identity Server and the Administration Console are installed on separate machines.

SAML 2.0: If you are using SAML 2.0, the metadata path is `/nidp/saml2/metadata`. For Site B in Figure 5-2, specify the following for SAML 2.0:

```
http://idp.siteb.novell.com:8080/nidp/saml2/metadata
```

SAML 1.1: If you are using SAML 1.1, the metadata path is `/nidp/saml/metadata`. For Site B in Figure 5-2, specify the following for SAML 1.1:

```
http://idp.siteb.novell.com:8080/nidp/saml/metadata
```

3c Click **Next** > **Finish** > **OK**.

3d Update the Identity Server.

Wait for the health status to return to green.

4 Continue with "Configuring Site B to Trust Site A as an Identity Provider" on page 58.

Configuring Site B to Trust Site A as an Identity Provider

The following instructions explain how to import the trusted root certificate and metadata of Site A into the configuration for Site B.

1 Log in to the Administration Console for Site B.

The configuration of Site B can be created in the same Administration Console as Site A; it cannot be configured to be a cluster member of Site A.

2 Import the trusted root certificate of Site A into the NIDP trust store of Site B.

2a Click **Devices** > **Identity Servers** > **Edit** > **Security** > **NIDP Trust Store**.

2b In the Trusted Roots section, click **Auto-Import From Server**, then fill the following fields:

Server IP/DNS: Specify the IP address or DNS name of Site A. For Site A in Figure 5-2, specify the following:

```
idp.sitea.novell.com
```

Server Port: Specify 8443.

2c Click **OK**, then specify an alias for the certificate (for example, SiteA).

You will get two certificate options: Root CA Certificate and

Server certificate. We recommend you to select Root CA Certificate.

2d Examine the trusted root that is selected for you.

If the trusted root is part of a chain, make sure you select the parent and all intermediate trusted roots.

2e Click **OK**.

The trusted root certificate of Site A is added to the NIDP trust store.

2f Click **Close**.

2g Click **Identity Servers** > **Update** > **OK**.

Wait for the health status to return to green.

3 Configure an identity provider for Site B.

3a Click **Identity Servers** > **Edit** > **Liberty** [or **SAML 2.0** or **SAML 1.1**].

3b Click **New**, select **Identity Provider**, then fill the following fields:

Name: Specify a name for the provider. If you plan on configuring more than one protocol, include the protocol as part of the name, such as SiteA_Liberty

Metadata URL: Specify the URL of the Liberty metadata on Site A. For Site A in Figure 5-2, specify the following:

```
http://idp.sitea.novell.com:8080/nidp/idff/metadata
```

This example uses port 8080 to avoid any potential certificate problems that occur when the Identity Server and the Administration Console are installed on separate machines.

SAML 2.0: If you are using SAML 2.0, the metadata path is `/nidp/saml2/metadata`. For Site A in Figure 5-2, specify the following for SAML 2.0:

```
http://idp.sitea.novell.com:8080/nidp/saml2/metadata
```

SAML 1.1: If you are using SAML 1.1, the metadata path is `/nidp/saml/metadata`. For Site B in Figure 5-2, specify the following for SAML 1.1:

```
http://idp.siteb.novell.com:8080/nidp/saml/metadata
```

3c Click **Next**.

3d To configure an authentication card, fill in the following:

ID: (Optional) Specify an alphanumeric number that identifies the card. If you need to reference this card outside of the Administration Console, you need to specify a value here. If you do not assign a value, the Identity Server creates one for its internal use.

Text: Specify the text that is displayed on the card to the user

Image: Specify the image to be displayed on the card. Select the image from the drop down list. To add an image to the list, click **Select local image**.

Login URL: (Conditional) If you are configuring an authentication card for SAML 1.1, specify an Intersite Transfer Service URL. For Figure 5-1 on page 54, specify the following value:

```
https://idp.sitea.novell.com:8443/nidp/saml/idpsend?PID=https://
idp.siteb.novell.com:8443/nidp/saml/metadata&TARGET=https://
idp.siteb.novell.com:8443/nidp/app
```

For more information, see "Specifying the Intersite Transfer Service URL for the Login URL Option" in the *NetIQ Access Manager Appliance 3.2 SP2 Identity Server Guide*.

Show Card: Determine whether the card is shown to the user. If this option is not selected, the card is only used when a service provider makes a request for the card. For this scenario, select this option.

Passive Authentication Only: Do not select this option.

3e Click **Finish** > **OK**.

3f Update the Identity Server.

Wait for the health status to return to green.

4 Continue with one of the following:

- If you are using Liberty or SAML 2.0, continue with "Verifying the Trust Relationship" on page 60.
- If you are using SAML 1.1, continue with "Configuring SAML 1.1 for Account Federation" on page 63.

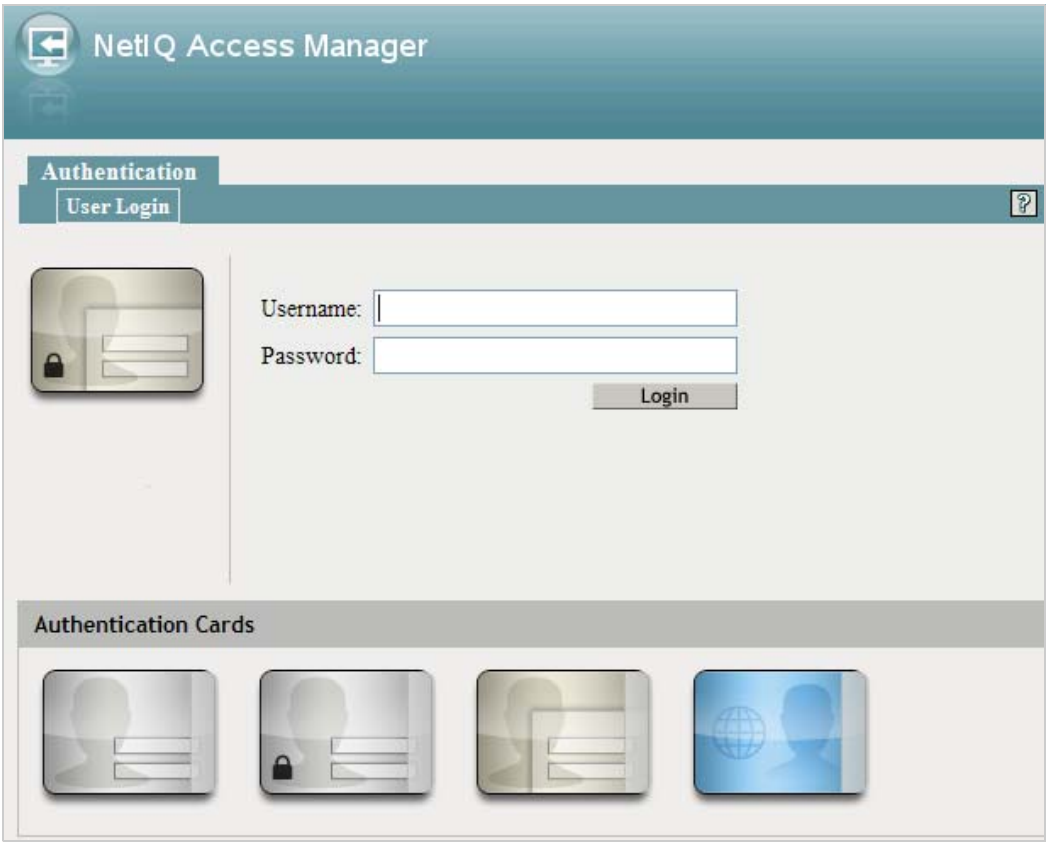
Verifying the Trust Relationship

Before continuing with federation configuration, you need to verify that Site A and Site B trust each other.

1 To test the trusted relationship, log in to the user portal of Site B. For Site B in Figure 5-2, specify the following:

```
https://idp.siteb.novell.com:8443/nidp/app
```

The following login screen appears.



In this configuration, the customizable image was used for the Liberty authentication card.

2 Click the Liberty (or SAML 2.0) authentication card.

You are directed to Site A for login, with the default card selected for you. A screen similar to the following appears:

3 Enter the credentials for a user from Site A.

The Federation consent prompt appears.

4 Click **Yes**.

You are returned to the login page for Site B.

5 Enter the credentials of a user from Site B that you want to federate with the user from Site A.

These two accounts are now federated. You can enter the URL to the user portal on Site A or Site B, and you are granted access without logging in again.

If you log out and log back in, the accounts are still federated, but you might be prompted for login credentials as you access resources on Site A and Site B. To enable a single sign-on experience, the Identity Server at Site A, the Identity Server at Site B, and the protected resources of the Access Gateways must be configured to share a contract.

6 To enable a single sign-on experience, continue with "Configuring User Authentication" on page 61.

Configuring User Authentication

The following instructions describe one way to enable single sign-on to the Identity Servers and Access Gateways in Figure 5-1 on page 54. It explains how to configure all sites to use the same contract. The instructions explain the following tasks:

- Selecting the contract for federation
- Configuring the contract at Site B to allow authentication at Site A
- Configuring Site A so its contract can satisfy the requirements of the contract at Site B
- Configuring Site A and Site B to use this contract as their default contract

To configure the contracts:

1 Log in to the Administration Console for Site B.

2 Configure the authentication request:

2a Click **Devices** > **Identity Servers** > **Edit** > **Liberty** [or **SAML 2.0**] > **[Name of Identity Provider]** > **Authentication Card** > **Authentication Request**.

2b (Liberty) Verify the settings of the following fields:

Allow federation: Make sure this option is selected. If this option is not selected, users cannot federate their accounts at Site A with an account at Site B.

After authentication: Make sure this option is selected. Enabling this option assumes that a user account exists at the service provider and that the account can be associated with a user's account at the identity provider.

During authentication: Make sure this option is selected. Enabling this option allows federation to occur when the user selects the authentication card of the identity provider.

2c (SAML 2.0) Verify the settings of the following fields:

Persistent: Select this option to set up a persistent relationship between the two accounts.

After authentication: Make sure this option is selected. Enabling this option assumes that a user account exists at the service provider and that the account can be associated with a user's account at the identity provider after authentication.

During authentication: Make sure this option is selected. Enabling this option allows federation to occur when the user selects the authentication card of the identity provider.

2d For **Requested By**, select **Use Contracts**.

2e (SAML 2.0) For Context Comparison, accept the default value of **Exact**.

2f In the **Authentication contracts** section, select the name of the contract used by the protected resources and move it to the **Contracts** section.

If the contract you require is not in the list, it has not been configured for federation. See Step 3.

2g Click **OK**, then update the Identity Server configuration.

3 (Conditional) Configure the contract at Site B to allow federation:

3a Click **Identity Servers** > **Edit** > **Local** > **Contracts**.

3b Record the URI for the contract you are using. This URI needs to exist as a contract on Site A. The name of the contract can be different at each site, but the URI must be the same.

NOTE: If site A only understands authentication class or type, select **Use Types** in the **Requested By** field and specify the authentication class in the **Allowable Class** field. Record the allowable class for the contract you are using. This allowable class should exist as a contract on site B. The name of the contract can be different at each site, but the allowable class must be the same.

3c Click the name of the contract.

3d Make sure the **Satisfiable by External Provider** option is selected.

3e Click **OK** twice, then update the Identity Server if you made any changes.

3f Return to Step 2 to select the contract.

4 Verify that Site A contains the same contract:.

4a Log in to the Administration Console for Site A.

4b Click **Identity Servers** > **Edit** > **Local** > **Contracts**.

4c Match the URI from Step 3b to a contract.

NOTE: Match the allowable class if you have selected **Use Types** in the **Requested By** field at site B.

If such a contract does not exist, you need to create it. For help, see "Configuring Authentication Contracts" in the *NetIQ Access Manager Appliance 3.2 SP2 Identity Server Guide*.

4d Click **OK**.

5 In the Administration Console for Site A, click **Identity Servers** > **Edit** > **Local** > **Defaults**.

6 For the Authentication Contract, select the name of the contract from Step 4c.

7 (Conditional) If you have multiple user stores, set the default contract for each user store.

8 Click **OK**, then update the Identity Server.

9 Test the configuration:

9a Enter the URL to the user portal of Site B.

9b Click the federated login link to Site A.

9c Enter the credentials for Site A and log in.

9d Enter the URL for a protected resource at Site B.

You are granted access without being prompted for credentials.

10 If you want to allow federated users to log in at Site A rather than using the card at Site B to redirect them to Site A, complete the following tasks:

10a In the Administration Console for Site B, click **Devices** > **Identity Servers** > **Edit** > **Local** > **Defaults**.

10b For the Authentication Contract, select the name of the contract whose URI matches the URI of the contract used by Site A.

10c Click **Liberty** [or **SAML 2.0]** > **[Name of Identity Provider]** > **Authentication Card** > **Authentication Request**.

10d In the **Options** section, enable the **Use automatic introduction** option.

This enables single sign-on to Site B when the user has already federated the accounts at the two sites.

10e Click **OK**, then update the Identity Server.

10f To test single sign-on, log in to the user portal on Site A, then enter a URL for a protected resource at Site B.

5.2.3 Configuring SAML 1.1 for Account Federation

SAML 1.1 does not support user-controlled federation, but you can configure it so that accounts that match are automatically federated. The Liberty and SAML 2.0 protocols allow users to federate accounts without sharing any common attributes, but the SAML 1.1 protocol requires that the user accounts need to share some common attributes in order for SAML 1.1 to match them and allow federation.

- "Configuring User Account Matching" on page 64
- "Configuring the Default Contract for Single Sign-On" on page 65
- "Verifying the Trust Relationship with SAML 1.1" on page 66

Configuring User Account Matching

When federating with SAML 1.1, the security of a user matching method depends upon the accuracy of the mapping. You need to select an attribute or attributes that uniquely identify the user at both Site A and Site B. The attributes must identify only one user at Site A and match only one user at Site B. If the attributes match multiple users, you have a security problem,

The following steps use the e-mail address of the user and the LDAP mail attribute to set up a matching rule that matches one user account at Site A with one user account at Site B. To securely use such a matching rule, you need to have a rule in place at both Site A and Site B to ensure that all users have unique e-mail addresses.

- "Configuring Site B for User Account Matching" on page 64
- "Configuring the Attribute for Sharing" on page 64
- "Configuring the Providers to Use the Shared Attribute" on page 65

Configuring Site B for User Account Matching

1. In the Administration Console of Site B, click **Devices** > **Identity Servers** > **Servers** > **Edit** > **SAML 1.1** > **[Identity Provider]** > **User Identification**.
2. For the **Satisfies contract** option, select the contract that you want to use for single sign-on.

   For this example, select **Secure Name/Password-Form**.
3. Select **Attribute matching**.

   The **Prompt for password on successful match** option is automatically selected. Leave this option enabled.
4. Click the **Define Attribute Matching Settings** icon.
5. Move the user store that you want to search for the attribute to the **User stores** list.
6. For the **User Matching Expression**, select **New User Matching Expression**.
7. Specify a name for the matching expression, such as email.
8. In **Logic Group 1**, click the **Add Attributes** icon, select **Ldap Attribute:mail [LDAP Attribute Profile]**, then click **OK**.

   The form allows you to create a very complex set of matching rules, with multiple conditions. This example uses one attribute, the simplest form of a matching expression.
9. Click **Finish**, then select your matching expression for the **User Matching Expression**.
10. Click **OK**.
11. Click **OK** twice, then update the Identity Server.
12. Continue with "Configuring the Attribute for Sharing" on page 64.

Configuring the Attribute for Sharing

1. In the Administration Console of the Site B (the service provider), click **Devices** > **Identity Servers** > **Shared Settings**.
2. Click **Attribute Sets**, then click **New**.
3. Specify a **Set Name**, such as email, then click **Next**.
4. Click **New**, then fill the **Add Attribute Mapping** options:

   Local attribute: Select **Ldap Attribute:mail [LDAP Attribute Profile]**.

   Remote attribute: Specify a name, such as email. Make sure you use the same remote name in the mapping for both Site B and Site A.

Leave the other options set to their default values.

5 Click **OK**, then click **Finish**.

Your newly created attribute mapping appears in the list of Attribute Sets.

6 Repeat Step 1 through Step 5 for Site A (the identity provider).

If Site A and Site B are imported into the same Administration Console, skip this step.

7 Continue with "Configuring the Providers to Use the Shared Attribute" on page 65.

Configuring the Providers to Use the Shared Attribute

You need to configure Site A to send the shared attribute with the authentication credentials, and you need to configure Site B to process the shared attribute that is included with the authentication credentials.

1 In the Administration Console for Site B, click **Devices** > **Identity Servers** > **Edit** > **SAML 1.1** > **[Name of Identity Provider]** > **Attributes**.

2 For the **Attribute set**, select the set name you created in "Configuring the Attribute for Sharing" on page 64.

3 Move the email attribute so that it is obtained at authentication.

4 Click **OK** twice, then update the Identity Server.

5 In the Administration Console for Site A, click **Devices** > **Identity Servers** > **Edit** > **SAML 1.1** > **[Name of Service Provider]** > **Attributes**.

6 For the **Attribute set**, select the set name you created in "Configuring the Attribute for Sharing" on page 64.

7 Move the email attribute so that it is sent with authentication.

8 Click **OK** twice, then update the Identity Server.

9 Continue with "Configuring the Default Contract for Single Sign-On" on page 65

Configuring the Default Contract for Single Sign-On

The Identity Servers at Site A and Site B need to use the contract you specified in your user matching expression to be the default contract for Site A, Site B, and the protected resources of the Access Gateway.

For the user matching expression contract, see Step 2 in "Configuring Site B for User Account Matching" on page 64.

To configure the default contracts for Site A and Site B:

1 In the Administration Console for Site B, click **Devices** > **Identity Servers** > **Edit** > **Local** > **Defaults**.

2 For the Authentication Contract, select the name of the contract used by the user matching expression.

3 Click **OK**, then update the Identity Server.

4 For Site A, repeat Step 1 through Step 3.

5 For the Access Gateway, review the contracts you have assigned to the protected resources:

5a In the Administration Console for Site B, click **Devices** > **Access Gateways** > **Edit** > **[Name of Reverse Proxy]** > **[Name of Proxy Service]** > **Protected Resources**.

5b For single sign-on, change the contract to match the contract for the user matching expression.

5c (Conditional) If you have multiple reverse proxies and proxy services, verify the contracts on all protected services that you want enabled for single sign-on.

5d Click **OK** to save your changes, then update the Access Gateway.

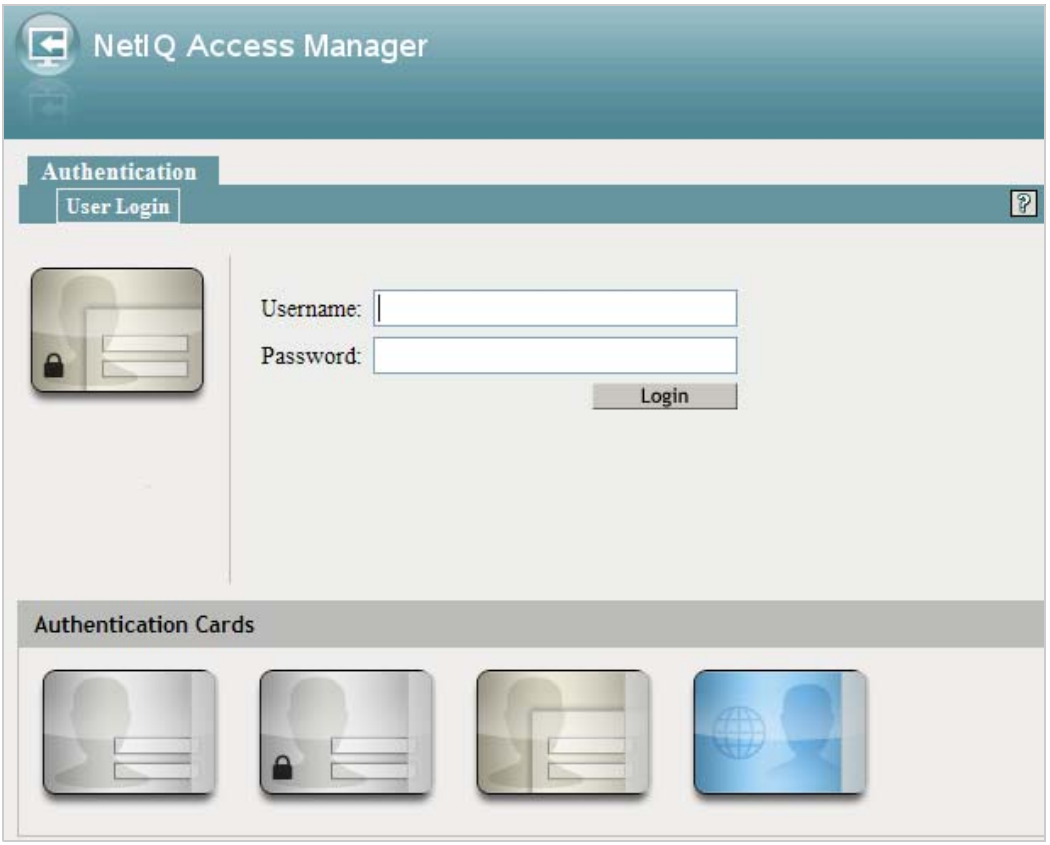
6 Continue with "Verifying the Trust Relationship with SAML 1.1" on page 66.

Verifying the Trust Relationship with SAML 1.1

1 To test the trusted relationship, enter the URL for the user portal of Site B. For Site B in Figure 5-2, you would specify the following:

```
https://idp.siteb.novell.com:8443/nidp/app
```

The following login screen appears:



Use the scroll bar to see all available cards.

2 Click the card you have configured for SAML 1.1 authentication.

You are directed to Site A for login.

3 Enter the credentials for Site A.

4 Enter the password for the user at Site B.

You are directed to the target page specified in the Login URL of the authentication card.

If you disabled the **Prompt for password on successful match** option on the User Identification page, the accounts are mapped without any user interaction.

5 (Conditional) If you receive an error, try one of the following:

- If you are not redirected to the target URL on Site B, verify the value you enter for the Login URL option. See Step 3d on page 59.
- If you receive an authentication error at Site B, verify the user matching setup. See "Configuring User Account Matching" on page 64.
- If you have enabled logging, open the logging file (`catalina.out` or `stdout.log`) and search for the error string. There should be additional information about the cause of the error in the error string entry as well as log entries before the error sting.

6 (Optional) If your protected resources on Site A and Site B use the same contract, enter the URLs of these resources.

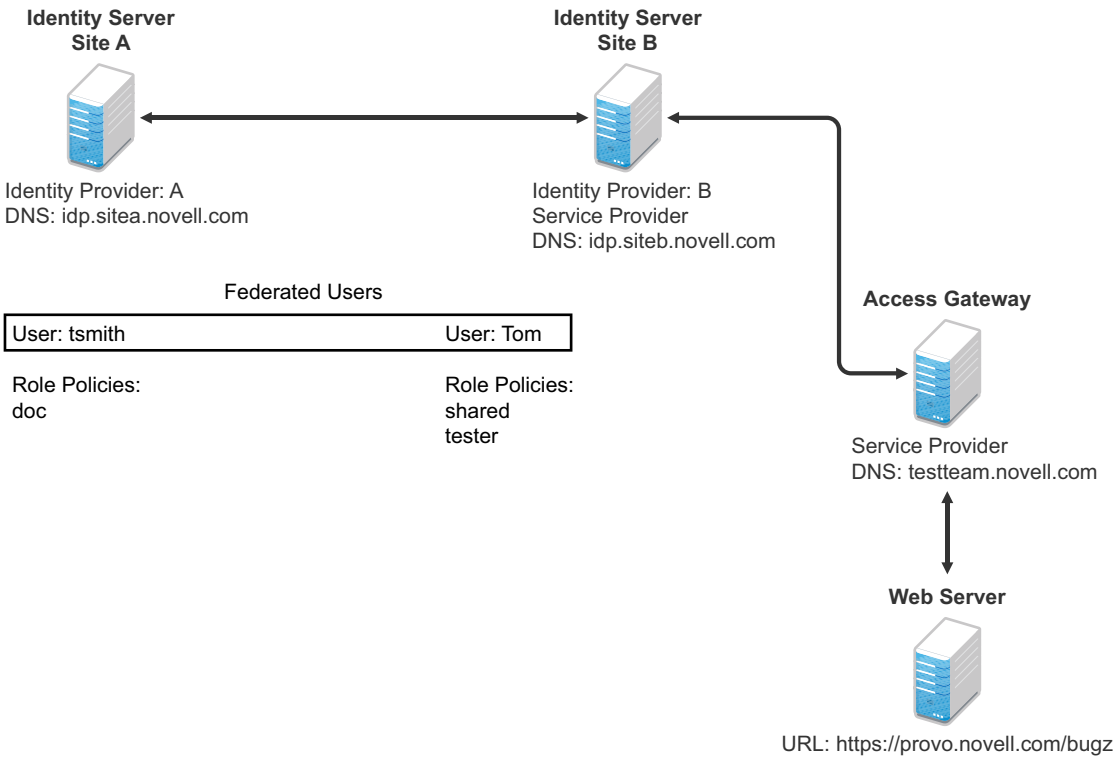
You are granted access without entering any additional credentials.

5.3 Sharing Roles

When two Identity Servers are configured to trust each other, one as an identity provider and the other as a service provider, they can be configured so that roles are shared. The following instructions are written for when both the identity provider and the service provider are NetIQ Identity Servers. If you are using a third-party identity or service providers, you need to modify the instructions.

Figure 5-3 illustrates a configuration where Identity Server of Site A is acting as an identity provider for Site B. When you configure the Identity Servers correctly, the Access Gateway can use the roles defined for the users of Site A in its policies.

Figure 5-3 *Two Federated Identity Servers*



The key to sharing roles is to set up the configuration so that the SAML assertion that the identity provider (Site A) sends to the service provider (Site B) contains the roles that the user has been assigned. Site B evaluates the roles and assigns them to the federated users at Site B. The Access Gateway can use these roles in its policy evaluations, and grant or deny access based on the assigned roles.

For example, when user tsmith authenticates to Site A, tsmith is assigned the role of doc. Tom, a user at Site B, is federated with the tsmith user. The doc role is shared with Site B, and Site B contains a policy that assigns users with the shared doc role to the tester role. The Access Gateway is configured with an Authorization policy that grants access to a resource when the requester is assigned the tester role. However, Tom does not have the qualifications at Site B to be assigned the tester role.

In this scenario, when Tom requests access to the protected resource at Site B, a login page with a federated link to Site A is displayed. If Tom selects to log in to Site A, Site A assigns him to the doc role. The doc role is sent with tsmith's authentication credentials to Site B. Site B evaluates the credentials and assigns Tom to the tester role because the following conditions are met:

- Tom is federated with tsmith.
- tsmith was assigned the doc role.
- The shared role and tester policies on Site B qualify the user to be assigned the tester role.

When the Access Gateway evaluates the credentials of Tom, Tom is granted access to the protected resource because he now has the tester role.

This section describes how to set up such a configuration. It assumes that the following have already been done:

- The trusted relationship between the identity provider and service provider is set up. For configuration instructions, see Section 5.2.2, "Establishing Trust between Providers," on page 57.
- The following policies have been created: the doc role policy at Site A, the tester role policy at Site B, and the Authorization policy (that uses the tester role) for the Access Gateway. For information about creating a Role policy, see Configuring a Role-Based Policy (https://www.netiq.com/documentation/netiqaccessmanager32/basicconfig/data/b6z0c3k.html#b6udq8w), and for the Authorization policy, see Assigning an Authorization Policy to Protect a Resource (https://www.netiq.com/documentation/netiqaccessmanager32_appliance/basicconfig/data/b6z0c3k.html#b6ug88w). The following instructions explain how to set up the shared policy.

This section explains how to configure Site A and Site B so that Site A shares its roles with Site B.

- Section 5.3.1, "Configuring Role Sharing," on page 69
- Section 5.3.2, "Verifying the Configuration," on page 72

5.3.1 Configuring Role Sharing

There are three major tasks for configuring role sharing. You need to configure a shared attribute for transferring the roles. You need to configure the identity provider and the service provider so that the role assignments can be added to the attribute and retrieved from the attribute. Finally, you need to create a shared Role policy for each role sent to the service provider. This policy defines how the role should be processed.

The following sections describe these configuration tasks:

- "Defining a Shared Attribute Set" on page 69
- "Obtaining the Role Assignments" on page 69
- "Configuring Policies to Process Received Roles" on page 70

Defining a Shared Attribute Set

1. In the Administration Console of the Site A (the identity provider), click **Devices** > **Identity Servers** > **Shared Settings**.
2. Click **Attribute Sets**, then **New**.
3. Specify a **Set Name**, such as role_sharing, then click **Next**.
4. Click **New** and fill the **Add Attribute Mapping** options:

   Local attribute: Select **All Roles**.

   Remote attribute: Specify a name, such as roles. Make sure you use the same remote name in the mapping for both the identity provider and the service provider.

   Leave the other options set to their default values.
5. Click **OK**, then click **Finish**.

   Your newly created attribute mapping appears in the list of Attribute Sets.
6. Repeat Step 1 through Step 5 on Site B (the service provider).
7. Continue with "Obtaining the Role Assignments" on page 69.

Obtaining the Role Assignments

1. To export the roles from the identity provider, log in to the Administration Console for the identity provider. (In Figure 5-3, this is Site A.)
   - **1a** Click **Devices** > **Identity Servers** > **Edit** > **Liberty** > **[Name of Service Provider]** > **Attributes**.

     If you are using SAML 2.0 or SAML 1.1 protocol, the steps are the same. You just need to click the appropriate tab after clicking **Edit**. The path is the same for these protocols.
   - **1b** Select the attribute set you created, then move **All Roles** so this attribute is sent with authentication.
   - **1c** Click **OK**.
   - **1d** Update the Identity Server of Site A.

2 To import the roles from the identity provider to the service provider, log in to the Administration Console for the service provider. (In Figure Figure 5-3, this is Site B.)

   2a Click **Devices** > **Identity Servers** > **Edit** > **Liberty** > **[Name of Identity Provider]** > **Attributes**.

   2b Select the attribute set you created, then move **All Roles** so this attribute is obtained with authentication.

   2c Click **OK**.

   2d Update the Identity Server of Site B.

   2e Continue with "Configuring Policies to Process Received Roles" on page 70.

Configuring Policies to Process Received Roles

For each role that is sent from Site A, you need to create a Role policy that specifies the role that should be activated on Site B. For example, suppose the tsmith user from Site A is assigned the doc role at authentication. You can create a Role policy on Site B that assigns the tester role to anyone with the doc role from Site A.

1 Log in to the Administration Console for Site B.

2 Click **Policies** > **Policies** > **New**.

3 Specify a name for the policy, select **Identity Server: Roles** for the type, then click **OK**.

4 In the **Condition Group 1** section, click **New**, then select **Roles from Identity Provider**.

5 (Conditional) If you have federated with more than one identity provider, select the provider. If you have federated with only one identity provider, the provider is selected for you.

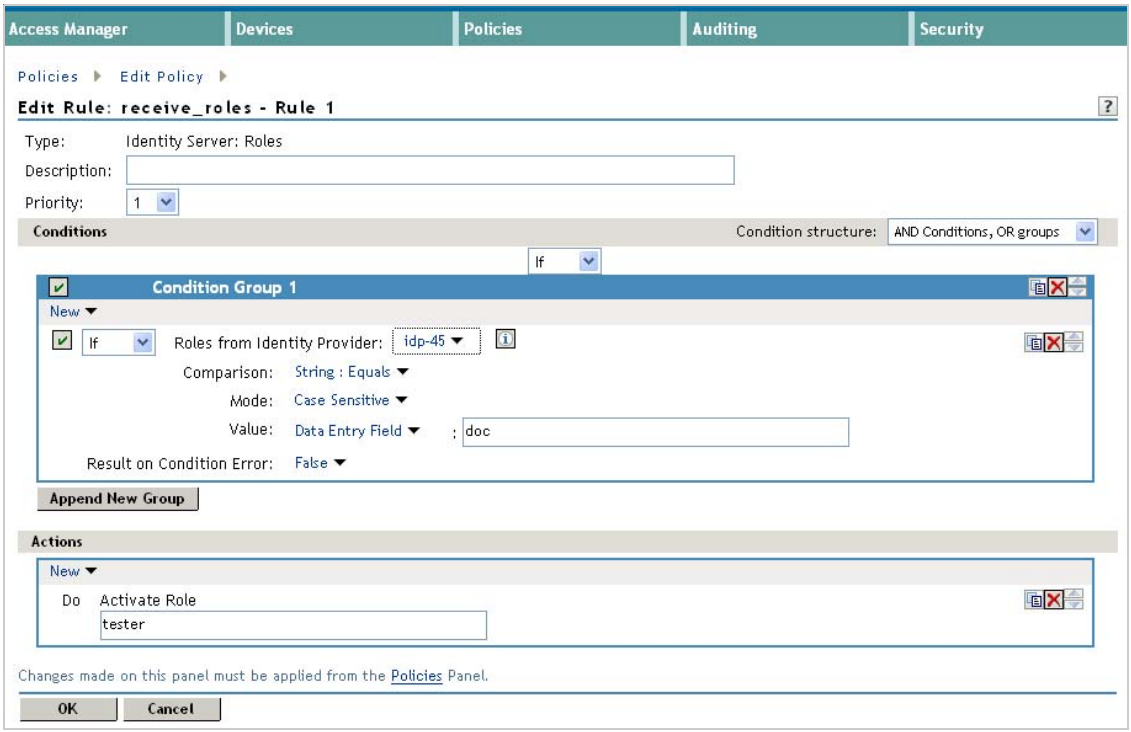
   In this example, you have federated with only the identity provider at Site A, and it is selected for you.

6 For the value, select **Data Entry Field**, then specify the name of a role that is assigned by Site A, for example doc.

   If you leave **Mode** set to **Case Sensitive**, make sure you specify the case correctly.

7 In the **Actions** section, specify the role to activate on Site B for the role received from Site A.

   Your policy should look similar to the following:



8 Click **OK** twice, then click **Apply Changes**.

9 To enable the role for the Identity Server, click **Identity Servers** > **Edit** > **Roles**.

10 Select the role, then click **Enable**.

11 (Optional) Repeat Step 2 through Step 10 for other roles assigned at Site A.

If you have other Role policies at Site A, you need to set up Role policies at Site B to have the roles activated. For example, if Site A had a Tester Role policy and you wanted users assigned to the Tester Role policy to also be assigned to the Tester Role policy at Site B, you could create a separate policy for this activation, or you could add an Or condition group with a value field of tester to the policy in Step 7. The policy would assign federated users who belonged to the doc or tester roles at Site A, to the tester role at Site B.

12 To test role sharing:

12a Enter the URL of a protected resource that requires a role for access. For the policy above, it would be a resource requiring the tester role.

12b Click the federated link to Site A.

12c Log in with the credentials of a user who is assigned the doc role.

You are granted access to the resource. If you are denied access, continue with Section 5.3.2, "Verifying the Configuration," on page 72 to discover the problem.

5.3.2 Verifying the Configuration

This section traces the role assignment from the Identity Server that assigns it to the user, through the Identity Server that receives the roles with the user's authentication assertion, to the policy evaluation. If you are having trouble, this should help you determine the source of the problem.

The following procedures refer to the configuration displayed in Figure 5-3, "Two Federated Identity Servers," on page 67. A tsmith user from Site A, who is assigned the doc role, is federated with a Tom user at Site B. Site B does not assign Tom the tester role. The Web server has been configured to protect the bugz site, which requires the tester role.

To verify the configuration:

1. Make sure policy logging is enabled on the identity provider and the service provider. Make sure that you enable at least Application logging at an Info level.

   For configuration procedures, see "Enabling Component Logging" in the *NetIQ Access Manager Appliance 3.2 SP2 Identity Server Guide*.

   You can access log files for downloading and viewing by clicking **Auditing** > **General Logging**.

2. Have a user access a resource that is protected by a policy requiring a role from Site A.

   For this trace, the tsmith user from Site A requests access to the bugz page. The user uses the federated link and logs in with the credentials of the tsmith user.

3. Verify that Site A is assigning the user the role.

   3a View the `catalina.out` file (Linux) or the `stdout.log` file (Windows) of the Identity Server at Site A.

   3b Search for the name of the role. You should find a line similar to the following:

   ```
   <amLogEntry> 2009-08-22T20:30:19Z INFO NIDS Application: AM#500105013:
   AMDEVICEID#C5F467BA50B009AC: AMAUTHID#DEEF6BEC3655DEB71CA56832DDDF866E:
   Authenticated user cn=tsmith,o=novell in User Store sitea-nids-user-store
   with roles doc,authenticated. </amLogEntry>
   ```

   If the role you need is not listed, look at the policy evaluation trace to discover why the user has not been assigned the role. For more information about how to understand role traces, see "Role Assignment Traces" in the *NetIQ Access Manager Appliance 3.2 SP2 Policy Guide*.

4. Verify that Site A is sending an authentication assertion to Site B.

   In the `catalina.out` file (Linux) or the `stdout.log` file (Windows) of the Identity Server from Site A, look for lines similar to the following:

   ```
   <amLogEntry> 2009-08-22T20:30:19Z INFO NIDS Application: AM#500105018:
   AMDEVICEID#C5F467BA50B009AC: AMAUTHID#DEEF6BEC3655DEB71CA56832DDDF866E:
   Responding to AuthnRequest with artifact
   AAPLsCVpfv3ha9Mpn+cUiCXcf3D63sc0QfscL5mZaaygHBKVOOh9aPSQ </amLogEntry>

   <amLogEntry> 2009-08-22T20:30:19Z INFO NIDS Application: AM#500105019:
   AMDEVICEID#C5F467BA50B009AC: AMAUTHID#F8B1C147EB3DDEFE9A3DB0827BA8E4A3:
   Sending AuthnResponse in response to artifact
   AAPLsCVpfv3ha9Mpn+cUiCXcf3D63sc0QfscL5mZaaygHBKVOOh9aPSQ </amLogEntry>
   ```

   If you do not see these types of entries, verify that you have configured Site A to send the roles. See "Obtaining the Role Assignments" on page 69.

5. Verify that Site B is receiving the SAML assertion with the roles.

   In the `catalina.out` file (Linux) or the `stdout.log` file (Windows) of the Identity Server from Site B, look for lines similar to the following:

```
<amLogEntry> 2009-08-22T20:30:19Z INFO NIDS Application: AM#500105020:
AMDEVICEID#488475009C6D3DDF: AMAUTHID#0FBA0CF7E41E6C7F9121DABB918D34F4:
Received and processing artifact from IDP -
AAPLsCVpfv3ha9Mpn+cUiCXcf3D63sc0QfscL5mZaaygHBKVOOh9aPSQ </amLogEntry>

<amLogEntry> 2009-08-22T20:30:19Z INFO NIDS Application: AM#500105021:
AMDEVICEID#488475009C6D3DDF: AMAUTHID#0FBA0CF7E41E6C7F9121DABB918D34F4:
Sending artifact AAPLsCVpfv3ha9Mpn+cUiCXcf3D63sc0QfscL5mZaaygHBKVOOh9aPSQ to
URL https://rholm.provo.novell.com:8443/nidp/idff/soap at IDP </amLogEntry>
```

The artifact ID should be the same as the artifact ID in Step 4.

If you do not see these types of entries, verify that you have configured Site B to receive the roles. See "Obtaining the Role Assignments" on page 69.

6 Verify that Site B is evaluating the received role assignments and activating the roles.

In the `catalina.out` file (Linux) or the `stdout.log` file (Windows) of the Identity Server from Site B, search for a policy evaluation for RolesFromIdentityProvider. You should find lines similar to the following:

```
~~CO~1~RolesFromIdentityProvider(6670):https://ipd.sitea.provo.novell.com:
8443/nidp/idff/metadata:TESTER,DOC,AUTHENTICATED~com.novell.nxpe.condition.
NxpeOperator@string-equals~(0):hidden-param:hidden-value:~~~True(69)

~~PA~ActionID_1203705845727~~AddRole~tester~~~Success(0)

<amLogEntry> 2009-08-22T20:30:20Z INFO NIDS Application: AM#500105013:
AMDEVICEID#488475009C6D3DDF: AMAUTHID#0FBA0CF7E41E6C7F9121DABB918D34F4:
Authenticated user cn=Tom,o=novell in User Store Internal with roles
tester,authenticated. </amLogEntry>
```

The policy evaluation shows that the condition evaluates to true and that the tester role is activated. Tom is the user that is federated with the tsmith user, and the entry shows that Tom has been assigned the tester role.

If you do not see a policy evaluation for RolesFromIdentityProvider, make sure you have created such a Role policy and that you have enabled it. See "Configuring Policies to Process Received Roles" on page 70.

7 If the use has been assigned the correct role, the last step is to verify how the embedded service provider evaluated the policy protecting the resource.

In the `catatina.out` file of the `ipd-esp` file for the Access Gateway, search for lines similar to the following for the authorization policy trace:

```
<amLogEntry> 2009-08-22T20:30:20Z INFO NIDS Application: AM#501102050:
AMDEVICEID#esp-2559E77C93738D15: AMAUTHID#BCF3CB40B51E8A0AF8582BEF762B4DDD:
PolicyID#65LN233O-KN19-1L7M-176M-P942LMN6P832: NXPESID#1411: AGAuthorization
Policy Trace:
 ~~RL~1~~~~Rule Count: 2~~Success(0)
 ~~RU~RuleID_1198874340999~Allow_Tester~DNF~~1:1~~Success(0)
 ~~CS~1~~ANDs~~1~~True(69)
 ~~CO~1~CurrentRoles(6660):no-param:TESTER,AUTHENTICATED~com.
novell.nxpe.condition.NxpeOperator@string-substring~SelectedRole
(6661):hidden-param:hidden-value:~~~True(69)
 ~~PA~1~~Permit Access~~~~Success(0)
 ~~PC~1~~Document=(ou=xpemlPEP,ou=mastercdn,ou=ContentPublisher
Container,ou=Partition,ou=PartitionsContainer,ou=VCDN_Root,ou=accessManagerCon
tainer,o=novell:romaContentCollectionXMLDoc),Policy=(Allow_Tester),Rule=(1::Ru
leID_1198874340999),Action=(Permit::1)~~~~Success(0)
 </amLogEntry>
```

If the PA line does not evaluate to Permit Access, then you need to review the Authorization policy and discover the conditions, other than the tester role, that must be met to permit access.

5.4 Setting Up Federation with Third-Party Providers

Setting up federation with providers other than NetIQ Identity Servers requires the same basic tasks as setting up federation with NetIQ Identity Servers, with some modifications.

When you set up federation with identity providers and service providers that are controlled by a single company, you have access to the Administration Consoles for both Identity Servers and know the admin credentials. When setting up federation with another company, additional steps are required.

- You need to negotiate with the other company and gain approval for federation because metadata must be shared and both sites require configuration. You need to negotiate a schedule for these configuration changes.
- The other site might not be using Access Manager for its identity or service provider. The basic tasks need to be modified to accommodate how that implementation shares metadata, authentication methods, and roles.
- Many SAML 1.1 providers do not support a metadata URL, and the data must be imported manually.

  For example, instead of sharing URLs that allow you to import metadata, you might need to share the actual metadata and paste it into the configuration. The NetIQ Identity Server validates the metadata of another identity provider or service provider; some implementations do not validate it. If the Identity Server determines that the metadata is invalid, you need to negotiate with the provider to send you metadata that has been validated.
- Most third-party providers do not support authentication cards and contracts. However, most do support either authentication types or authentication URIs. You can use either of these to map from their authentication procedure to an Identity Server authentication contract.

For sample implementations with third-party providers that explain the modifications that were required to set up the federation, see the following:

- "Integrating Novell's Access Manager with Shibboleth's IDP Server" (http://www.novell.com/communities/node/6943/integrating-novells-access-manager-shibboleths-idp-server)
- "Integrating Google Apps and Novell Access Manager using SAML2" (http://www.novell.com/communities/node/8645/integrating-google-apps-and-novell-access-manager-using-saml2)
- "SAML 1.1 with Concur" (http://www.novell.com/coolsolutions/appnote/19673.html)

5.5 External Attribute Source Policy Examples

You can use an External Attribute Source policy to retrieve attributes from external sources. You can create shared secrets from this policy. This shared secret then can be used in configuring other policies or can be used by the Identity Servers in their attribute sets to retrieve attributes from external sources.

An External Attribute Source policy must be enabled and configured before using the policy for retrieving the attributes from external sources.

For more information about how to create an External Attribute Source policy, see "Creating External Attribute Source Policies" in the *NetIQ Access Manager Appliance 3.2 SP2 Policy Guide*.

This section describe the usages of the External Attribute Source policy with the help of the following scenarios:

- Section 5.5.1, “Scenario 1,” on page 75
- Section 5.5.2, “Scenario 2,” on page 77

For information about sample codes for these examples, see Access Manager SDK Sample Code.

5.5.1 Scenario 1

e_Health is a Web portal for doctors. e_Health uses Med_Association as an external identity provider to verify whether the user is a doctor and obtain the user's professional code and specialization. Med_Association retrieves these details with the help of the NetIQ Identity Server.

Med_Association completes the following steps:

1. Write an External Attribute data extension class and use the required attribute to retrieve the professional code and specialization of user.

   For more information about data extension class, see “Adding Policy Extensions” in the *NetIQ Access Manager Appliance 3.2 SP2 Policy Guide*.

   For more information about data extension example code, see The Policy Extension API in the NetIQ Access Manager 3.2 Developer Kit guide.

2. Create an External Attribute Source policy for the data extension.

   For more information about how to import the data extension class and configure the External Attribute Source policy in the Identity Server, see “Configuring an External Attribute Source Policy” in the *NetIQ Access Manager Appliance 3.2 SP2 Policy Guide*.

3. Define a shared secret for the professional code and specialization.

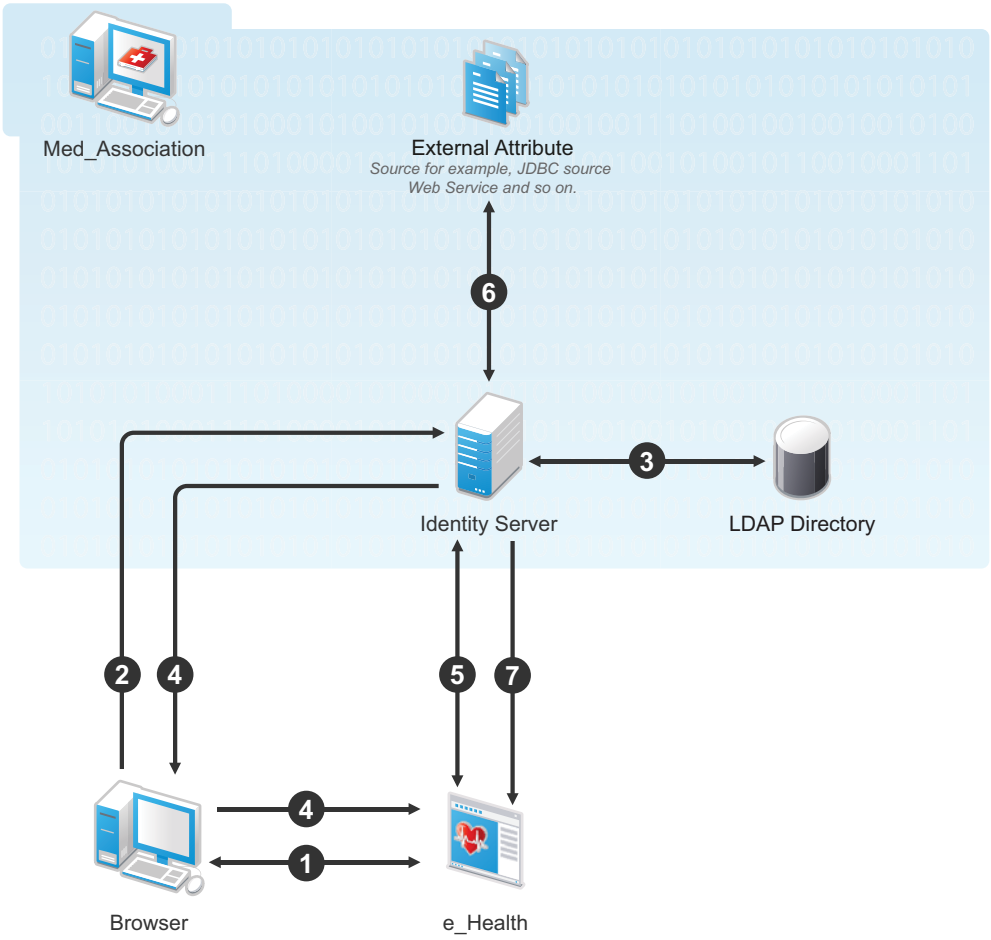
   For more information, see “Adding Custom Attributes” in the *NetIQ Access Manager Appliance 3.2 SP2 Identity Server Guide*.

4. Configure this shared secret for a service provider to be sent with authentication.

   For more information, see “Configuring the Attributes Sent with Authentication”in the *NetIQ Access Manager Appliance 3.2 SP2 Identity Server Guide*.

5. The retrieved details that are professional code and specialization are sent to e_Health.

The following diagram illustrates this scenario:



Workflow:

1. User requests for access to e-Health through browser.
2. e_Health redirects the user's browser to the NetIQ Identity Server at Med_Association for authentication.
3. User logs in with providing credentials. User is authenticated with LDAP.
4. On the successful authentication, the Identity Server sends the assertion to e_Health.
5. e_Health verifies the assertion with Med_Association by using the back channel communication.
6. After verification, the NetIQ Identity Server retrieves the attributes (professional code and specialization) from external sources (for example, database) by using the External Attribute Source policy.
7. The Identity Server returns the response containing professional code and specialization in a shared secret attribute. If the user is not a doctor, external source returns null values in the shared secret attribute in the response.

   e_Health grants access to the user if it receives valid values for the attributes in the authentication response else it denies the access.

5.5.2 Scenario 2

Company XYZ is a customer of NetIQ Access Manger. The employees of this company get authenticated to the Identity Server. Each employee's mail attribute is retrieved from the user store. XYZ wants only user name part of the email address to be displayed on the Home page after authentication. This can be achieved by using the External Attribute Source policy.

XYZ completes the following steps:

1. Write an External Attribute data extension class and use the mail attribute as the parameter to the class.

   For more information about data extension class, see "Adding Policy Extensions"in the *NetIQ Access Manager Appliance 3.2 SP2 Policy Guide*.

2. In the data extension class, read the email address and parse the name identifier in it and return as an attribute. For more information about data extension example code and example code for this scenario, see The Policy Extension API in the NetIQ Access Manager 3.2 Developer Kit guide.

3. Define a shared secret for the name field of the email address.

   For more information, see "Adding Custom Attributes" in the *NetIQ Access Manager Appliance 3.2 SP2 Identity Server Guide*.

4. Create an External Attribute Source policy for the data extension.

   For more information about how to import the data extension class and configure the External Attribute Source policy in the Identity Server, see "Configuring an External Attribute Source Policy" in the *NetIQ Access Manager Appliance 3.2 SP2 Policy Guide*.

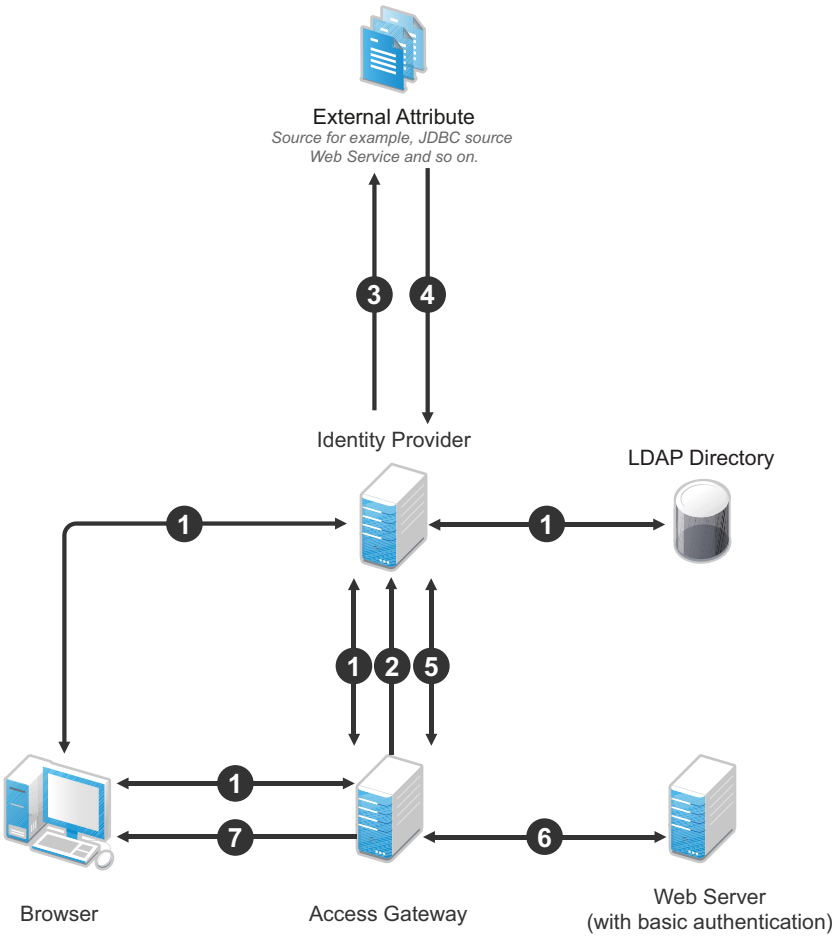
5. Create an Identity Injection policy.

   For more information, see "Creating Identity Injection Policies"and Configuring a Custom Header Policy in the NetIQ Access Manager Appliance 3.2 SP2 Policy Guide.

6. The Identity Server sends the user ID part of email address to the Access Gateway.

   In turn, the Access Gateway or service provider sends this attribute to the configured Web server. For example, John is an employee of XYZ. He provides his email address, john@mail-domain.com, as his user name. After authentication, only John will be displayed on the Home page.

The following diagram illustrates this scenario:



Workflow:

1. User (through the browser) is requesting for a resource. The Access Gateway determines whether it is a protected resource and redirects the request to the Identity Server for authentication. The Identity Server authenticates with the LDAP servers and provides the assertion details to the Access Gateway. In turn, the Access Gateway verifies the assertion details.
2. The Home page in the resource is configured to display the user ID that has to be retrieved from the Identity Server.
3. The Identity Server determines whether the attributes can be retrieved from the external source. The Identity Server will send the required details to the external source (in this example, an email address).
4. The external source returns the data. In this example, user ID part of the email address.
5. The Identity Server sends the data that it has obtained from the external source to the Access Gateway.
6. The Access Gateway sends the data to the Web server.
7. The Web server returns the resource.

5.6 Step up Authentication Example

This section discusses a Step up Authentication example for the Identity Server initiated SSO.

For more information about Identity Server initiated SSO, see "Using the Intersite Transfer Service" in the *NetIQ Access Manager Appliance 3.2 SP2 Identity Server Guide*.

Setup: Let us assume that:

- NetIQ Access Manager is acting as the identity provider.
- The following three contracts in the identity provider are configured:
  - name password basic contract with Authentication level as 10
  - name password form contract with Authentication level as 20
  - secure name password contract with Authentication level as 30

  NOTE: Enable the Satisfiable by a contract of equal or higher level option for contracts with authentication level 10 or 20 to avoid prompting for authentication when a user is already authenticated against the contract with level 30.
- The name password form contract for a service provider named SP_A is configured in the identity provider.

  For more information about creating and configuring the contracts, see "Configuring Authentication Contracts"in the *NetIQ Access Manager Appliance 3.2 SP2 Identity Server Guide*.

Configuration: Complete the following steps:

1. In the NetIQ Identity Server, configure the service provider as a trusted provider.

   For more information, see "Managing Trusted Providers"in the *NetIQ Access Manager Appliance 3.2 SP2 Identity Server Guide*.
2. In the service provider, configure the NetIQ Identity Server as a trusted provider.

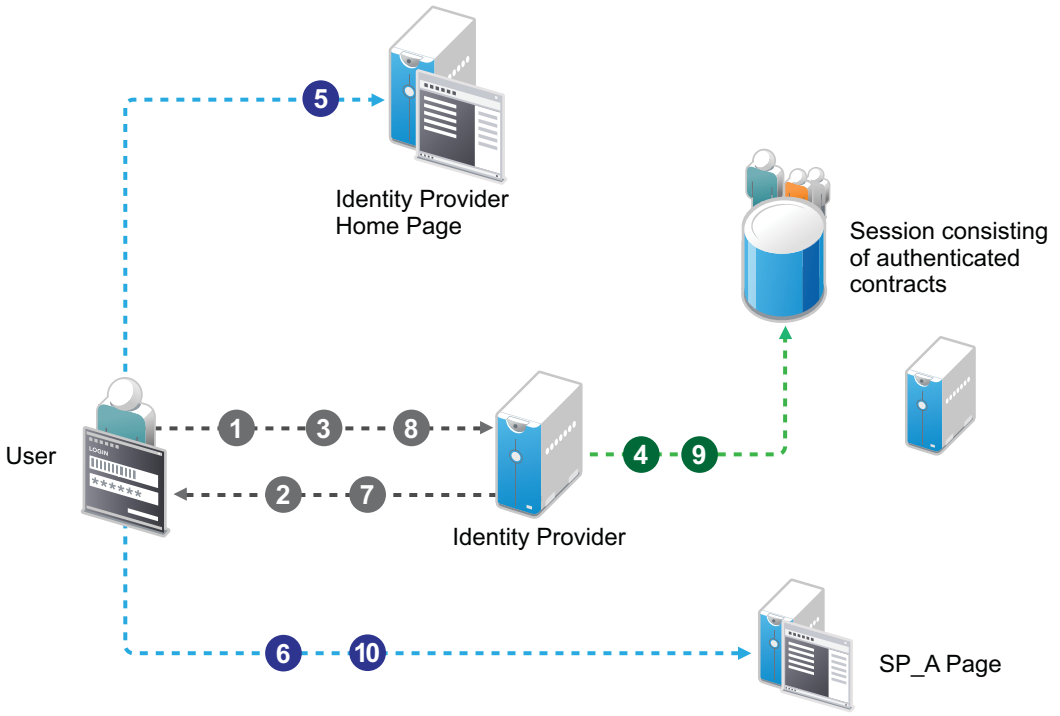
   For more information, see "Managing Trusted Providers"in the *NetIQ Access Manager Appliance 3.2 SP2 Identity Server Guide*.
3. In the NetIQ Identity Server, configure the service provider with the required authentication contracts.

For information about how to configure a service provider, see "Defining Options for Liberty or SAML 2.0 Service Provider", "To Define Options for Liberty Service Provider "and"Defining Options for SAML 1.1 Service Provider"in the *NetIQ Access Manager Appliance 3.2 SP2 Identity Server Guide*.

Results: The following are the four possible scenarios:

- If the user was authenticated with the name password basic contract before making an Intersite Transfer Service request to SP_A, the identity provider will step up to the name password form authentication.
- If the user was authenticated with the name password form contract before making an Intersite Transfer Service request to SP_A, the identity provider will not ask for the authentication.
- If the user was authenticated with the secure name password contract before making an Intersite Transfer Service request to SP_A, the identity provider will not ask for the authentication.
- If the user is not authenticated while making an Intersite Transfer Service request to SP_A, the identity provider will step up to the name password form authentication.

The following diagram illustrates the workflow:



Workflow:

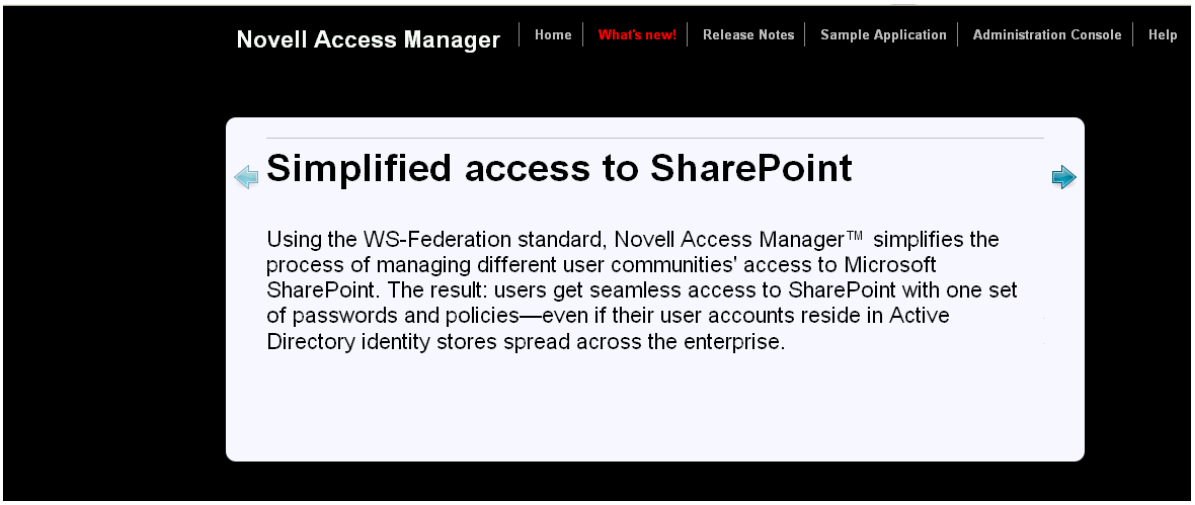
1. User tries to authenticate in the identity provider.
2. User is prompted to authentication using the Name Password Basic contract.
3. User enters the credentials.
4. The Name Password Basic contract is authenticated in the identity provider and added to the user session.

   The Name Password Basic contract is the default contract in the identity provider.
5. User logs into the identity provider.
6. User makes an Intersite Transfer Service request to SP_A.
7. The identity provider prompts for the authentication using the Name Password Form contract.
8. User enters the credentials.
9. The Name Password Form contract is authenticated in the identity provider and added to the user session.
10. User is redirected to SP_A.

6 Access Manager Appliance Portal

The sample application that comes by default with the Access Manager Appliance showcases the various Access Manager features. The landing page is illustrated in Figure 6-1 on page 81. Ensure that you remove the landing portal in the production environment. Instructions for removing this portal are given on the landing page.

Figure 6-1 *Access Manager Appliance' Landing Portal Page*



The Access Manager Appliance is configured by default to allow access to this first landing page and the default policies are created and assigned to protect the other pages.

The example Web pages are designed to help network administrators understand the basic concepts of Access Manager Appliance by installing a relatively simple implementation of the software. The example serves as a primer for a more comprehensive production installation of Access Manager Appliance.

- Section 6.1, "Access Manager Appliance Overview and Prerequisites," on page 81
- Section 6.2, "Accessing the Sample Web Portal," on page 83
- Section 6.3, "Understanding the Policies Used in the Sample Portal," on page 84

6.1 Access Manager Appliance Overview and Prerequisites

This section discusses the concepts involved in installing Access Manager Appliance to protect the Web portal:

- Section 6.1.1, "Overview," on page 82
- Section 6.1.2, "Prerequisites," on page 83

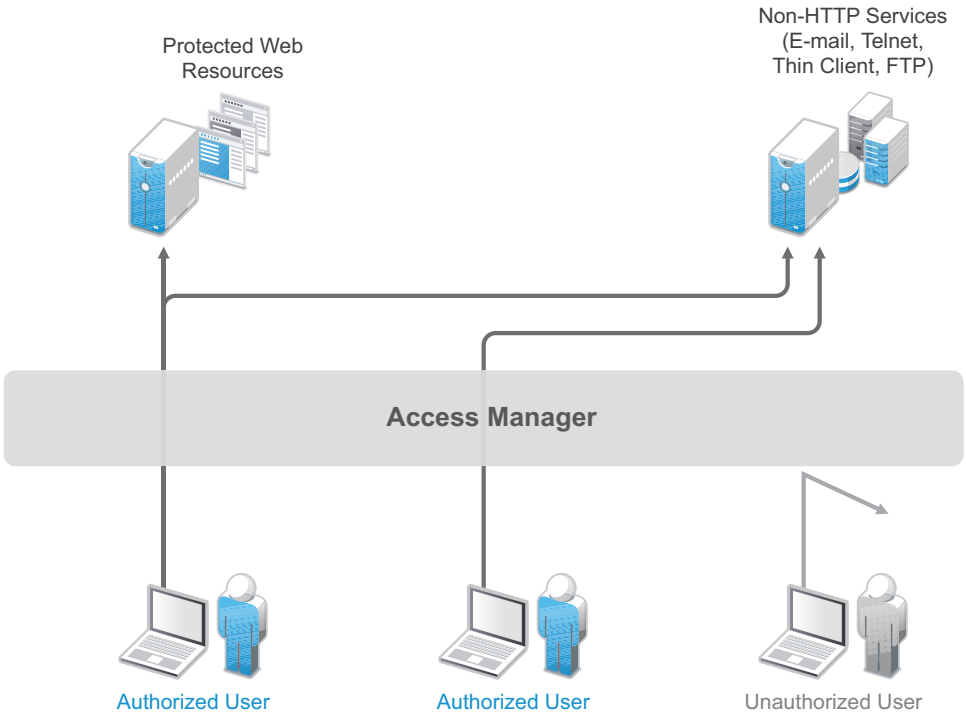
After you go through this portal, you should understand the basic features of Access Manager Appliance and know how the software is used to protect your own Web servers and applications.

6.1.1 Overview

Access Manager Appliance offers a simplified deployment model. The Access Manager components are deployed as an appliance in a single-box form factor. For more information, see "Installing the Access Manager Appliance"in the *NetIQ Access Manager Appliance 3.2 SP2 IR1 Installation Guide*.

The primary purpose of Access Manager Appliance is to protect resources by allowing access only to users you have authorized. You can control access to Web (HTTP) resources as well as traditional server-based (non-HTTP) resources. As shown in the following illustration, those users who are authorized to use the protected resources are allowed access, while unauthorized users are denied access.

The following diagram illustrates how the default web portal is integrated with the Access Manager Appliance.



Access Manager Appliance secures your protected Web resources from Internet hackers. The addresses of the servers that host the protected resources are hidden from both external and internal users. The only way to access the resources is by logging in to Access Manager Appliance with authorized credentials.

You can see the IP address of the Access Gateway installed in the Access Gateways window. The health of the configured Access Gateway is Green. The published DNS name to access your sample web portal site, in this example uses *namapp.com*. This DNS name resolves the IP address set up as the listening address. When you edit the *Reverse Proxy / Authentication*, you can see that it is already configured.

In the *Identity Server Cluster* option, the configuration assigned to the Identity Server that is the default `IDP-Cluster` is displayed. This establishes the trust relationship between the Access Gateway and the Identity Server that is used for authentication. In the *Reverse Proxy List* `NAM-RP` which is the default reverse proxy is listed.

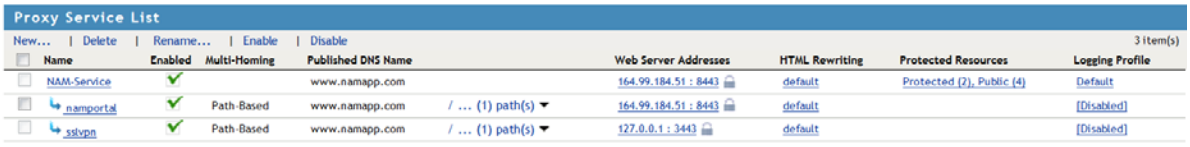
6.1.2 Prerequisites

Before starting with the default web portal, you must perform the following tasks:

- ❒ Enable pop-ups on a Mozilla Firefox browser or Internet Explorer browser (7.x or above) for managing and configuring the Access Manager Appliance components.
- ❒ Install the NetIQ Access Manager Appliance as described in the *NetIQ Access Manager Appliance 3.2 SP2 IR1 Installation Guide*.

6.2 Accessing the Sample Web Portal

You can access the sample web portal by going to the portal web site, in this example, *www.namapp.com/portal*. This is because the **namportal** is already configured with the published DNS name www.namapp.com and the Multi-Homing Path-Based proxy service is defined as /portal as in the figure below.

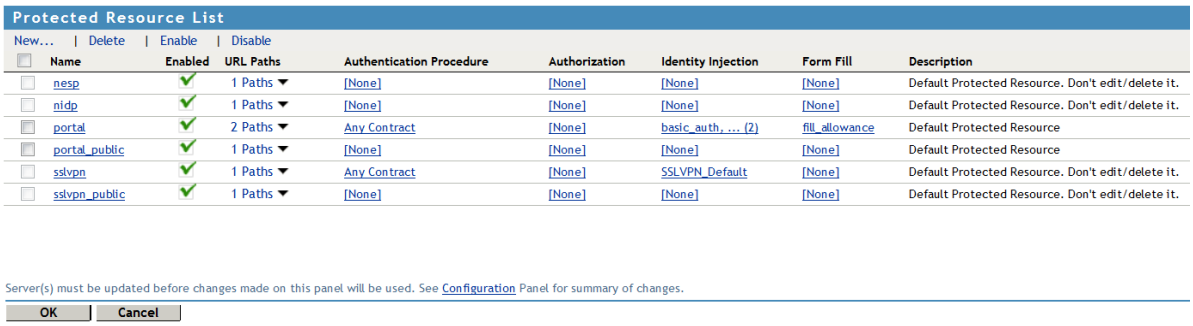


Protected resource details are displayed in the Protected Resource List as in the figure Figure 6-2 on page 83 below. You can see that the portal_public and sslvpn_public are public resources and do not have an authentication procedure. You can access these pages without any credentials from the following URLs:

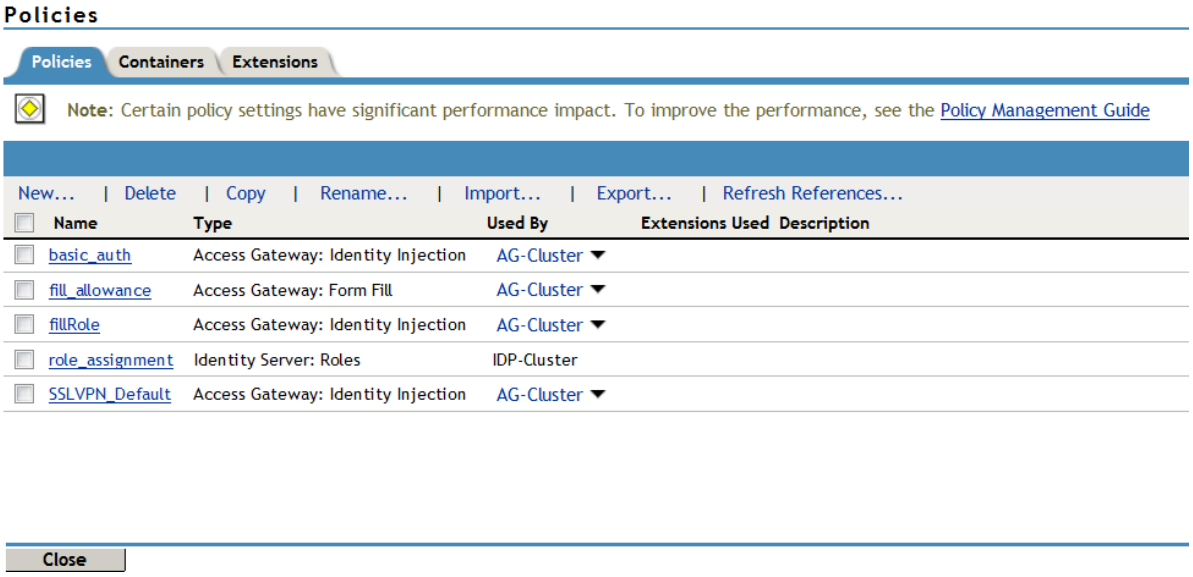
- (http://www.namapp.com/portal) takes you to the landing page of the web portal.
- (http://www.namapp.com/sslvpn/heartbeat) checks the SSL VPN status.

The default protected resources in this example are /portal/payinfos/* and /portal/users/* that have an associated authentication procedure. For example if you want to access the portal go to (http://www/namapp.com/portal) and click on **Sample Application** on the portal page. You will be asked for credentials. By default Access Manager creates two sample users Alice and Bob with password novell.

Figure 6-2 *Protected Resource List*



6.3 Understanding the Policies Used in the Sample Portal



The sample portal site is configured for authentication and role based authorization.

Access Manager Appliance uses an Identity Server Role policy to assign roles to logged in users. In the sample portal Identity Server with a policy named role_assignment Manager and Employee are defined. A user Alice is assigned with role Manager and Employee. Another user Bob is assigned with role Employee. The users of role Employee and Manager can see and edit their own as well as an employee's basic information. Payroll information of each user is a protected information. A user who is assigned the role of Employee cannot see the pay information of other users, unless assigned the role of Manager.

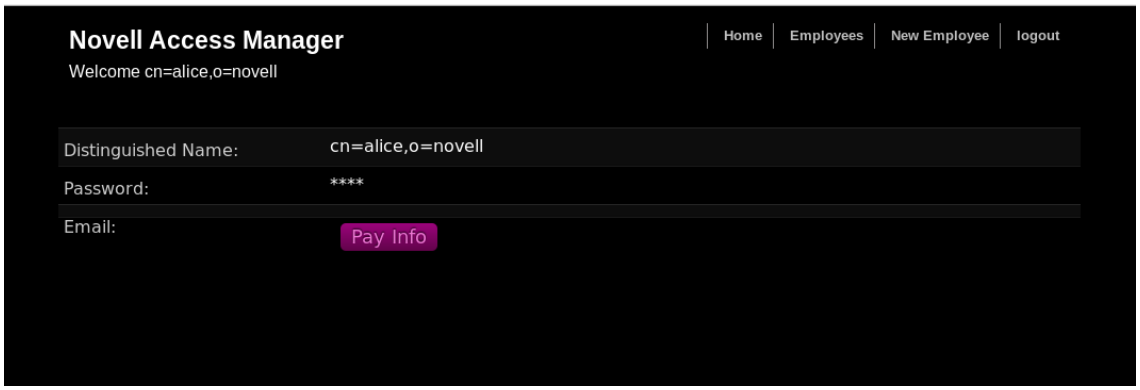
Access Manager Appliance uses authorization policy to define access control. Role Based Access Control can conveniently assign a user to a particular job function or set of permissions within an enterprise. Access Manager Appliance enables you to assign roles to users, based on attributes of their identity, and then associate policies with the roles. In designing your own actual production environment, you need to decide which roles you need (such as, sales, administrative, payroll, and accounting). You can create Role policies that assign the roles to your users, and then create Authorization and Identity Injection policies that use the roles to control access.

Access Manager uses the Identity Injection policy for single sign-on to a web resource using the HTTP header, for example, HTTP authentication. There are Identity Injection policies configured with basic_auth and fillRole which are used for single sign-on to the portal. basic_auth Identity Injection policy will inject authentication header with LDAP User DN and LDAP Password. The DN Format used is LDAP, for example, cn=alice,ou=Payroll,o=Novell. Fillrole injects the defined name and value, in this example Roles into the custom header. The main page of the sample payroll site displays the user's login name.

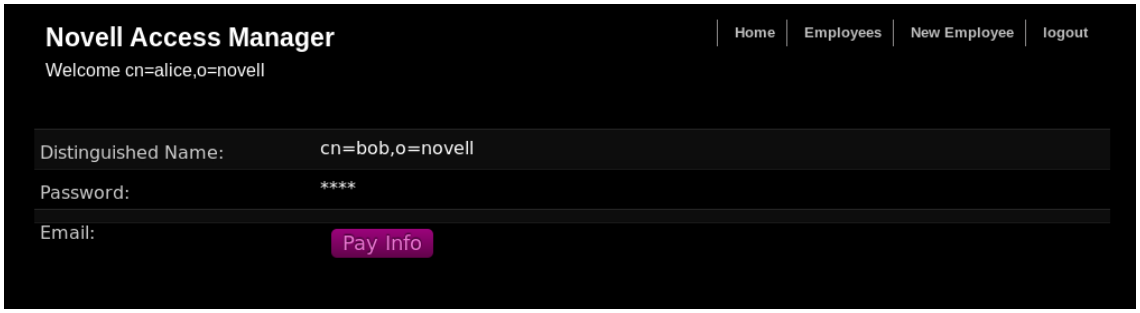
Access Manager uses the Form Fill policy to fill the forms from the Web server. A default Form Fill policy, fill_allowance is defined. The **Input Field Name** payinfo.allowances under **Fill Options** is defined with the value 10000. When you edit the pay info field, the **Allowances** field is automatically filled with this value. Any request without basic authentication headers and the required role will be forbidden.

You can use the sample application available to understand the roles by following the procedure below:

1. Login to the portal page for example, https://www.namapp.com/portal and click on **Sample Application**.
2. Login with the username alice. The following login page is displayed with the published DNS name alice. Alice can access her pay information. If the user belongs to payroll, the **Pay Info** button is displayed on the page.



3. Click on **Employees**. Alice can access Bob's pay information because Alice is assigned the manager role. Click **show** against the DNS name, in this example, Bob and click **Pay Info**.



4. Click **pay edit** to edit the pay of the employees. The **Allowances** field is automatically filled as defined in the Form Fill policy. You can edit the pay information and save your changes.
5. Click on **New Employee** to create a new employee.

NOTE: If you login as Bob, you cannot create a new employee or access the pay information of other employees and will get a Forbidden error as Bob is not assigned a Manager role.
