DEPLOYMENT GUIDE
Version 1.0

Deploying the BIG-IP Access Policy Manager v10.2.1 with Oracle Access Manager

Table of Contents

Configuring the BIG-IP APM for WebGate Reverse Proxy and Oracle Access Manager

Prerequisites and configuration notes 1
Product versions and revision history 2
Configuration example 2

Configuring the BIG-IP LTM 4
Creating the health monitor 4
Creating the pool 4
Creating the profiles 6
Creating the virtual server 8
Configuration checkpoint 10

Configuring the BIG-IP APM 12
Creating an Authentication Source 12
Creating the SSO configuration 13
Creating an Access Profile 14
Editing the Access Profile with the Visual Policy Editor 16

Modifying the Oracle configuration 18
Modifying the Oracle Authentication Rule 18

Enabling OAM and APM integration on the BIG-IP LTM virtual server 20
Modifying the virtual server to use the Access Policy 20

Optional: Using an iRule to enable or disable the Access profile 21

Configuring the BIG-IP APM for WebGate Reverse Proxy and Oracle Access Manager v10.2.1

Welcome to the F5 deployment guide for the BIG-IP Access Policy Manager (APM) and Oracle Access Manager. This guide describes how to configure the BIG-IP APM for Oracle Access Manager when you are looking to replace a WebGate Proxy farm with APM.

Oracle Access Manager helps enterprises create greater levels of business agility, ensure seamless business partner integration, and enable regulatory compliance. Through an innovative, integrated architecture Oracle Access Manager uniquely combines identity management and access control services to provide centralized authentication, policy-based authorizations, and auditing with rich identity administration functionality such as delegated administration and workflows.

For more information on Oracle Access Manager, see ***www.oracle.com/technology/products/id_mgmt/coreid_acc/index.html***.

Prerequisites and configuration notes

The following are prerequisites and configuration notes for this implementation:

- The WebGate Agent behind the BIG-IP APM may be disabled on the Application Web Tier servers.
- The default behavior of the BIG-IP APM is to protect access to ALL of the resources on the backend application servers (recommended). If you wish to only protect certain resources, as defined in your OAM policy, please refer to *Optional: Using an iRule to enable or disable the Access profile*, on page 21.
- It is assumed that you have Administrator privileges to your OAM installation. This is required, as you need to make minor modifications to your policy. For more information, see *Modifying the Oracle configuration*, on page 18.
- It is also assumed that your OAM policies are properly configured, such as authentication and authorization failures. The BIG-IP APM relies on the OAM server for defined behaviors, otherwise the flow/connection will be dropped for an undefined behavior.
- For more configuration options on the BIG-IP Access Policy Manager, see the Configuration Guide for BIG-IP Access Policy Manager, available on Ask F5 (***https://support.f5.com/***).

Product versions and revision history

Product and versions tested for this deployment guide:

Product Tested	Version Tested
BIG-IP APM	10.2.1
Oracle Identity Management/	11.1.1.1.0
Oracle Access Manager	10.1.4.3.0

Revision history:

Document Version	Description
1.0	New deployment guide for 10.2.1

Our Oracle Identity Management 11gR1 implementation was deployed according to the *Oracle® Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management 11g Release 1 (11.1.1) Part Number E12035-02*.

Configuration example

In this guide, we demonstrate an architecture where Oracle Access Manager provides authentication and authorization services to an application. Instead of authenticating users directly at the application layer with the WebGate agent or via a farm of WebGate Proxies, BIG-IP APM is used to perform the authentication and enforce authorization. Allowing APM to offload the WebGate functionality simplifies the OAM deployment by eliminating WebGate Agents from the application servers and consolidating the proxy layer onto the network infrastructure.

In this example, BIG-IP is first configured to provide access to the application. Once this configuration is completed and tested, the addition of OAM functionality can be added to provide restricted access to the application.

Figure 1 shows a logical configuration example before the BIG-IP APM has been implemented, and a BIG-IP Local Traffic Manager is directing traffic to the WebGate Proxy. Figure 2 shows the logical configuration example after the BIG-IP APM has been implemented.

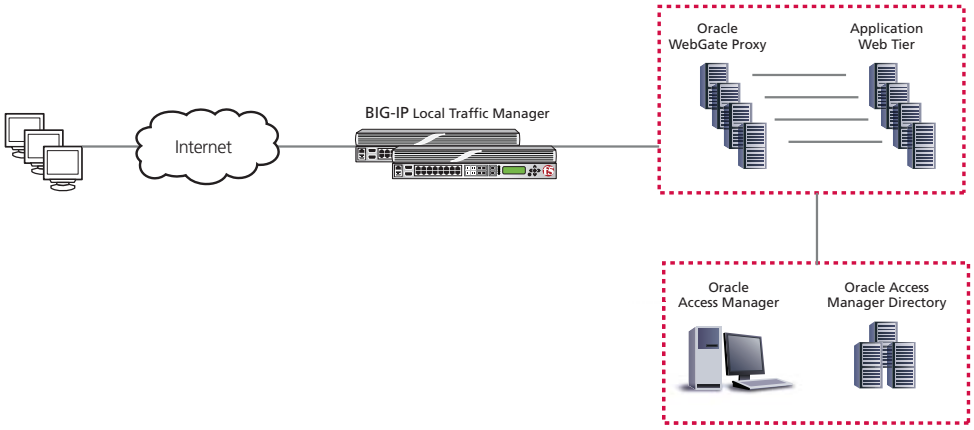

Figure 1 *Logical configuration example before the BIG-IP APM*

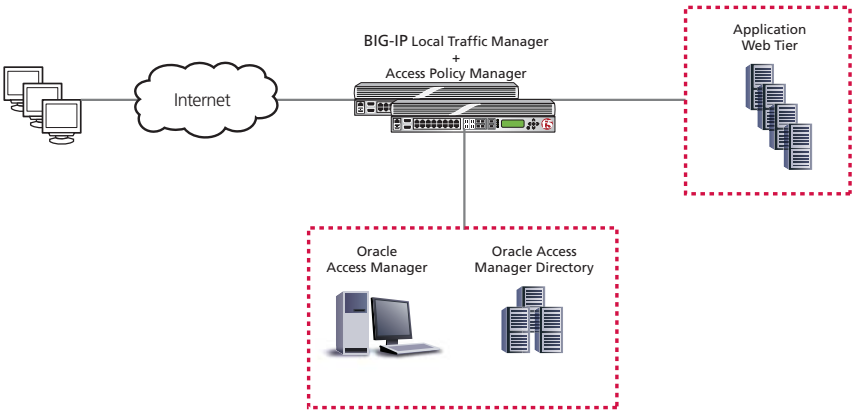

Figure 2 *Logical configuration example including the BIG-IP APM*

Configuring the BIG-IP LTM

First, we configure the BIG-IP LTM for Oracle Application web tier.

Note

If you are already using the BIG-IP LTM for your Oracle Web tier, you likely already have the following BIG-IP LTM objects configured. We recommend you review this section to ensure you have all relevant objects, and then continue with ***Configuring the BIG-IP APM****, on page 12.*

Creating the health monitor

The next step is to set up a health monitor for the Web servers. This procedure is optional, but very strongly recommended. In our example, we create an HTTP health monitor.

To configure a HTTP health monitor

1. On the Main tab, expand **Local Traffic**, and then click **Monitors**.
2. Click the **Create** button. The New Monitor screen opens.
3. In the **Name** box, type a name for the Monitor.
   In our example, we type **ora11g-web-monitor**.
4. From the **Type** list, select **HTTP**.
5. In the Configuration section, in the **Interval** and **Timeout** boxes, type an Interval and Timeout. We recommend at least a 1:3 +1 ratio between the interval and the timeout. In our example, we use a **Interval** of **30** and a **Timeout** of **91**.
6. In the **Send String** box, leave the default string: **GET /\n\n**.
7. The rest of the settings are optional, configure as appropriate for your implementation.
8. Click the **Finished** button.

Creating the pool

You must create a pool on the BIG-IP LTM system for the Oracle web servers.

To create the Oracle pool

1. On the Main tab, expand **Local Traffic**, and then click **Pools**. The Pool screen opens.
2. Click the **Create** button. The New Pool screen opens.
3. In the **Name** box, type a name for your pool. In our example, we use **ora11g-web-pool**.

4. In the **Health Monitors** section, select the name of the monitor you created in the *Configuring the BIG-IP LTM* section, and click the Add (**<<**) button. In our example, we select **ora11g-web-monitor**.
5. From the **Load Balancing Method** list, choose your preferred load balancing method (different load balancing methods may yield optimal results for a particular network). In our example, we select **Least Connections (node)**.
6. Leave the Priority Group Activation **Disabled**.
7. In the New Members section, make sure the **New Address** option button is selected.
8. In the **Address** box, add the server(s) hosting your applications. In our example, we type **10.133.15.60**.
9. In the **Service Port** box, type the service number you want to use for this device, or specify a service by choosing a service name from the list. In our example, we type **80**.
10. Click the **Add** button to add the member to the list.
11. Repeat steps 8-10 for each one of the servers.
12. Click the **Finished** button.

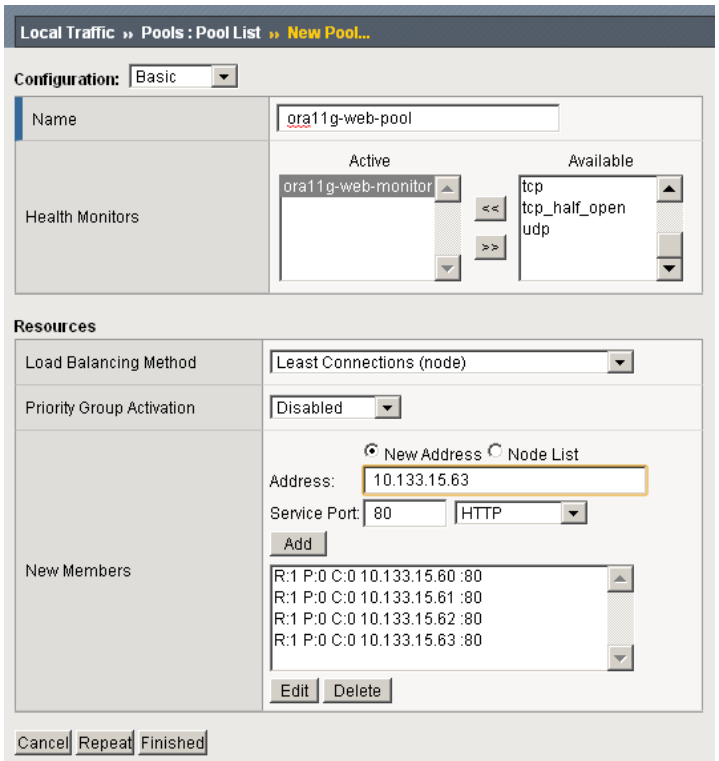

Figure 3 *BIG-IP pool configuration*

Creating the profiles

The next task to create the profiles. A ***profile*** contains user-configurable settings for controlling the behavior of a particular type of network traffic. Using profiles enhances your control over managing network traffic, and makes traffic-management tasks easier and more efficient.

Creating the HTTP profile

In this procedure, you create an HTTP profile. You must create an HTTP profile (or use the default profile) for this configuration to function.

To create a new HTTP profile

1. On the Main tab, expand **Local Traffic**, and then click **Profiles**.
2. Click the **Create** button. The New HTTP Profile screen opens.
3. In the **Name** box, type a name for this profile. In our example, we type **ora11g-web-http**.
4. From the **Parent Profile** list, select **http**.
5. Modify any of the settings as applicable for your network. In our example, we leave the settings at their default levels.
6. Click the **Finished** button.

Creating the SSL profile

If you are using the BIG-IP system to offload SSL, the next task is to create the client SSL profile. This profile contains SSL certificate and Key information for offloading SSL traffic.

If you are not using the BIG-IP system to offload SSL, continue with *Creating persistence profiles*, on page 7.

First we import the certificate and key (for this Deployment Guide, we assume that you already have obtained the required SSL certificates, but they are not yet installed on the BIG-IP LTM system. If you do not have a certificate and key, see the BIG-IP documentation).

To import a key or certificate

1. On the Main tab, expand Local Traffic.
2. Click **SSL Certificates**. This displays the list of existing certificates
3. In the upper right corner of the screen, click **Import**.
4. From the **Import Type** list, select the type of import (**Certificate** or **Key**).
5. In the **Certificate** (or **Key**) **Name** box, type a unique name for the certificate or key.

6. In the **Certificate** (or **Key**) **Source** box, choose to either upload the file or paste the text.
7. Click **Import**.
8. If you imported the certificate, repeat this procedure for the key.

The next task is to create the SSL profiles that uses the certificate and key you just imported.

Configuring the Client SSL profile

Use the following procedure to create the Client SSL profile. This profile is not necessary if you are not offloading SSL traffic on the BIG-IP system.

To create a new Client SSL profile

1. On the Main tab, expand **Local Traffic**, click **Profiles**, and then, on the Menu bar, from the **SSL** menu, select **Client**.
2. Click the **Create** button.
3. In the **Name** box, type a name for this profile. In our example, we type **ora11g-web-clientssl**.
4. In the Configuration section, click a check in the **Certificate** and **Key** Custom boxes.
5. From the **Certificate** list, select the name of the Certificate you imported in the *Importing keys and certificates* section.
6. From the **Key** list, select the key you imported in the *Importing keys and certificates* section.
7. Click the **Finished** button.

Creating persistence profiles

The final profiles we create are Persistence profiles. In this case, we create two persistence profiles; a default and a fallback persistence profile. Because we are using HTTP cookie insert persistence as our default mode, we need the fallback mode in case the user's device does not accept cookies.

Creating the Cookie Persistence profile

The first persistence profile we create is the Cookie Persistence profile. In this profile there are some optional settings you can configure, such as the method of cookie persistence and the expiration.

To create a new cookie persistence profile

1. On the Main tab, expand **Local Traffic**, and then click **Profiles**. The HTTP Profiles screen opens.
2. On the Menu bar, click **Persistence**.

3. Click the **Create** button.
4. In the **Name** box, type a name for this profile. In our example, we type **ora11g-web-cookie**.
5. From the **Persistence Type** list, select **Cookie**.
   The configuration options for cookie persistence appear.
   Make sure the Parent Profile is set to **Cookie**.
6. Modify any of the other settings as applicable for your network.
7. Click the **Finished** button.

Creating the Fallback Persistence profile

Now we configure the fallback persistence profile. In our example, we use Source Address Affinity for the fallback persistence type.

To create a new fallback persistence profile

1. On the Main tab, expand **Local Traffic**, click **Profiles**, and then on the Menu bar, click **Persistence**.
2. Click the **Create** button.
   The New Persistence Profile screen opens.
3. In the **Name** box, type a name for this profile. In our example, we type **ora11g-web-source**.
4. From the **Persistence Type** list, select **Source Address Affinity**.
   The configuration options for Source Address Affinity persistence appear.
5. Modify any of the settings as applicable for your network. In our example, we leave the settings at their default levels.
6. Click the **Finished** button.

Creating the virtual server

The next task is to create the virtual server to which users connect to the Oracle Application Web tier.

To create the virtual server

1. On the Main tab, expand **Local Traffic**, and then click **Virtual Servers**. The Virtual Servers screen opens.
2. Click the **Create** button. The New Virtual Server screen opens.
3. In the **Name** box, type a name for this virtual server. In our example, we type **ora11g-web-vs**.
4. In the **Destination** section, select the **Host** option button.
5. In the **Address** box, type the IP address for this virtual server. In our example, we type **10.133.15.101**.

6. In the **Service Port** box:

   a) If you are NOT using the BIG-IP system to offload SSL, type **80**, or select **HTTP** from the list.

   b) If you are using the BIG-IP system to offload SSL, type **443**, or select **HTTPS** from the list.

7. From the HTTP Profile list, select the HTTP profile you created in the *Creating the HTTP profile* section. In our example, we select **ora11g-web-http**.

8. If you are using the BIG-IP system to offload SSL, from the **SSL Profile (Client)** list, select the SSL profile you created in the *Configuring the Client SSL profile* section. In our example, we select **ora11g-web-clientssl**.

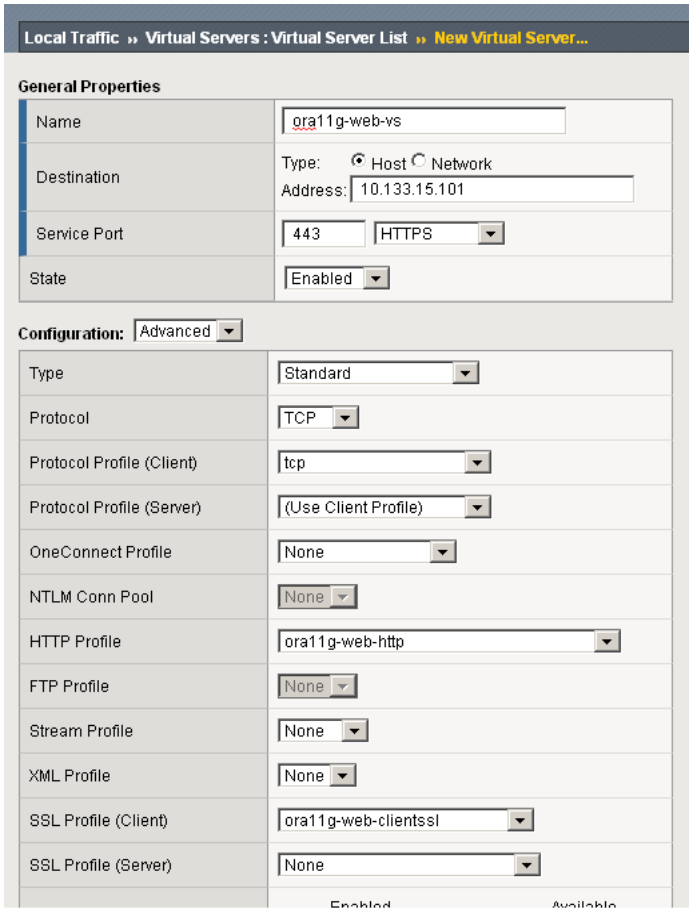

Figure 4 *Virtual server configuration (truncated)*

9. From the **Default Pool** list, select the pool you created in *Creating the pool*. In our example, we select **ora11g-web-pool**.

10. From the **Default Persistence Profile** list, select the profile you created in *Creating the Cookie Persistence profile*, on page 7. In our example, we select **ora11g-web-cookie**.
11. From the **Fallback Persistence Profile** list, select the profile you created in *Creating the Fallback Persistence profile*, on page 8. In our example, we select **ora11g-web-source**.
12. Click the **Finished** button.

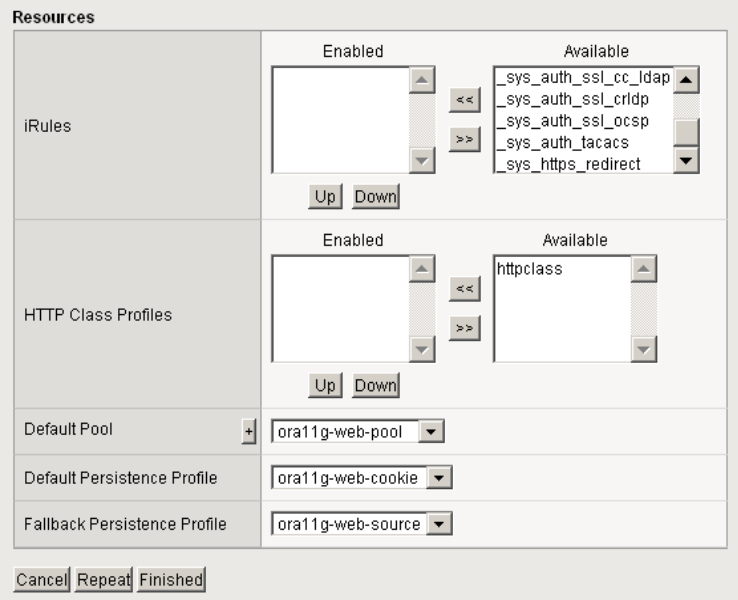

Figure 5 *Resources section of the virtual server configuration*

Configuration checkpoint

With the BIG-IP LTM configuration for the Oracle Application Web tier complete, traffic should transparently flow through the BIG-IP LTM to the Oracle resources. Before continuing the configuration in this guide, we recommend you confirm you can reach your Oracle resources using the BIG-IP LTM virtual server with a web browser.

If you cannot reach the Oracle resources, check the IP address on the BIG-IP virtual server, as well as the pool member IP addresses.

To check the pool member status:

1. From the Navigation pane, under Local Traffic, click **Pools**.
2. Click the name of the Oracle pool (**ora11g-web-pool** in our example).

3. On the Menu bar, click **Members**. You see a list of members. Each member should have a green circle as in the following image. If they are not green, check the specific device(s).

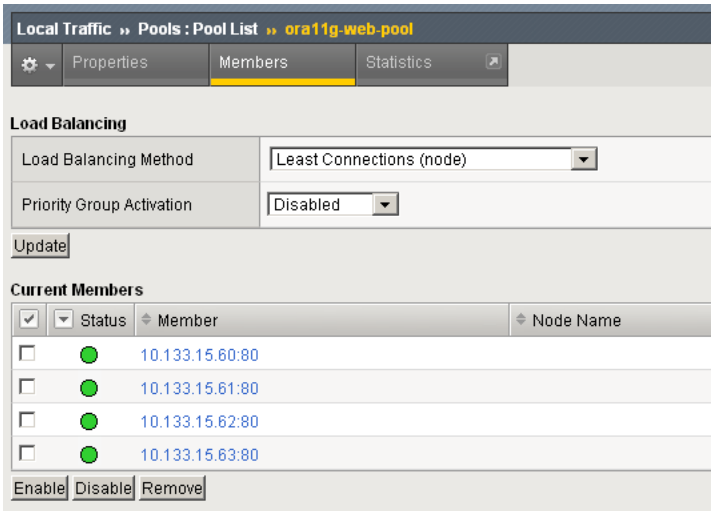

Figure 6 *Pool member status*

Configuring the BIG-IP APM

Use the following procedures to configure the BIG-IP APM for Oracle Access Manager.

Creating an Authentication Source

The first task is to create an Authentication Source that specifies details for connecting to the Access Server that your Oracle Access Manager installation uses.

To create an AAA server

1. On the Main tab, expand **Access Policy**, and then click **AAA servers**.
2. Click the **Create** button.
3. In the **Name** box, type a name for this profile. In our example, we type **ora11g-oam-aaa**.
4. From the **Type** list, select **Oracle Access Manager**.
5. In the **Access Server Name** box, type the name of your Access Server. In our example, we type **AccessServer_OAM0**.
6. In the **Access Server Hostname** box, type the FQDN (fully qualified domain name) of the Access server. In our example, we type **idm-oam0-11g.oracle.siterequest.com**.
7. In the **Access Server Port** box, type the appropriate port. In our example, we type **6023**.
8. In the **Access Gate Name** box, type the name. In our example, we type **IDMEDG_AG**.
9. In the **Access Gate Password** and **Verify Password** boxes, type the password.
10. From the **Transport Security Mode** box, select **Open** or **Simple**. In our example, we select **Simple**.
    If you select **Simple**, two Passphrase boxes appear. Type the applicable passphrase in both boxes.

Note

This mode must match the configured mode of the Access Gate you are connecting to within the OAM server.

11. Click **Finished** (see Figure 7, on page 13).

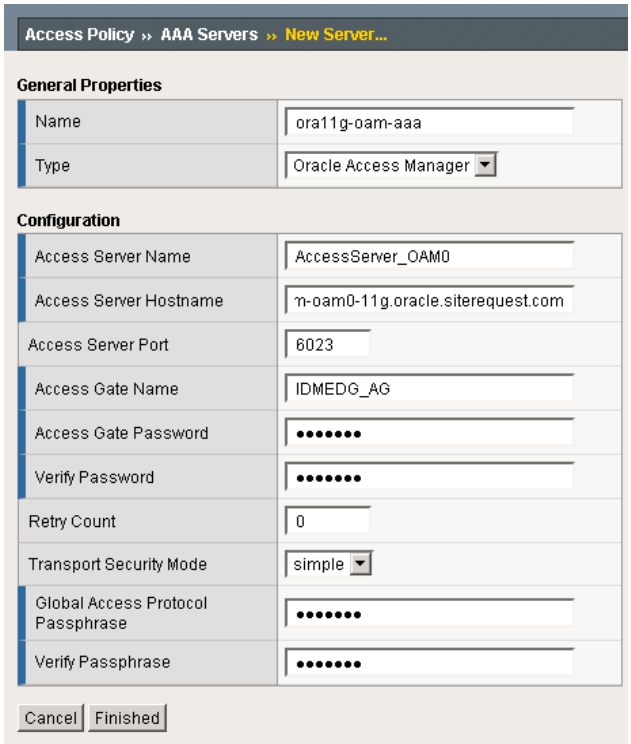

Figure 7 *BIG-IP APM AAA Server configuration*

Creating the SSO configuration

The next task is to create a Single Sign-On Configuration that defines the credentials that are cached.

To create the SSO configuration

1. On the Main tab, expand **Access Policy**, and then click **Web Applications**.
2. Click the **Create** button.
3. In the **Name** box, type a name for this profile. In our example, we type **ora11g-oam-sso**.
4. From the **SSO Method** list, select **None**.
5. In the **Username Source** box, type the user name source. In our example, we leave the default: **session.sso.token.last.username**.
6. In the **Password Source** box, type the user name source. In our example, we leave the default: **session.sso.token.last.password**.
7. From the **Access Management Method** list, select **Oracle Access Management**.

8. From the **Oracle Access Management Server** list, select the AAA server you created in *Creating an Authentication Source*, on page 12. In our example, we select **ora11g-oam-aaa** (see Figure 8).
9. Click **Finished**.

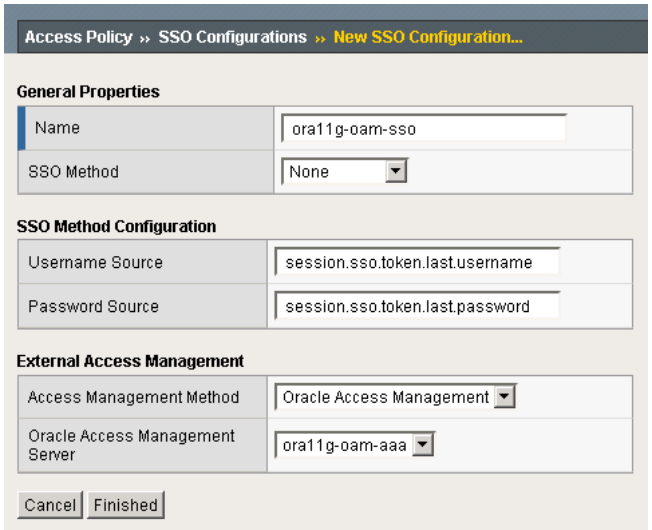

Figure 8 *SSO configuration*

Creating an Access Profile

The next task is to create an Access Profile and a Visual Policy which provides an antivirus check, a logon page, and SSO Credential mapping.

To create an Access Profile

1. On the Main tab, expand **Access Policy**, and then click **Access Profiles**.
2. Click the **Create** button.
3. In the **Name** box, type a name for this profile. In our example, we type **ora11g-oam-access**.
4. In the Settings section, configure the options as applicable for your configuration. In our example, we leave all of the settings at their defaults. Note that depending on licensing, the number of concurrent users may be limited. The other timeouts are administrative choices.
5. In the Configuration section, from the **SSO Configurations** list, select the name of the SSO Configuration you created in *Creating the SSO configuration*, on page 13. In our example, we select **ora11g-oam-sso**.

6. In the **Domain Cookie** box, type the domain in which the virtual server is located. This is the domain that users will connect to when accessing the Virtual Server. In our example, we type **siterequest.com**.

◆ Note

*It is strongly recommended that this domain setting be the same as the domain configured as the **Primary HTTP Cookie Domain** in your Access Gate Configuration.*

7. If your virtual server will be using HTTP instead of HTTPS, ensure the **Secure Cookie** box is *not* checked. If you are using HTTPS, make sure the Secure Cookie box is checked.
8. All other settings are optional, configure as applicable for your configuration. See Figure 9 for our settings.
9. Click **Finished**.

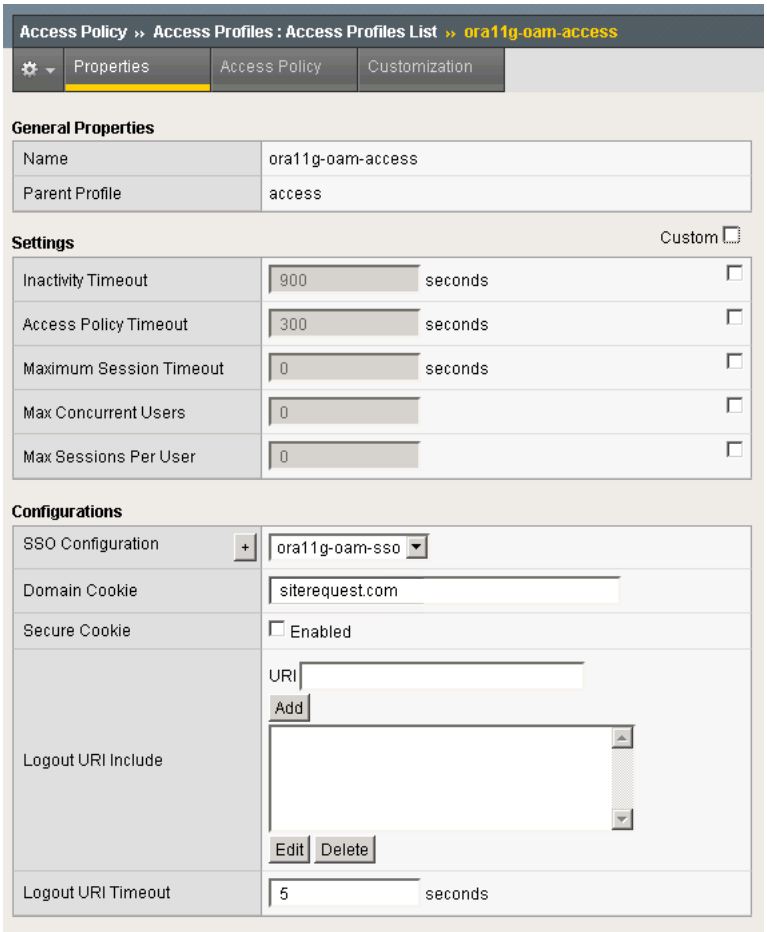

Figure 9 *Oracle Access Policy (truncated)*

Editing the Access Profile with the Visual Policy Editor

The next task is to edit the Access Policy using the Visual Policy Editor (VPE). The VPE is a powerful visual scripting language that offers virtually unlimited options in configuring an Access Policy.

For additional or more sophisticated authentication and policy options, see the ***Configuration Guide for BIG-IP Access Policy Manager***, available on Ask F5 (***https://support.f5.com/***).

To edit the Access Profile

1. On the Main tab, expand **Access Policy**, and then click **Access Profiles**.
2. Locate the Access Profile you just created, and in the Access Policy column, click **Edit**.
   The Visual Policy Editor opens in a new window.
3. Click the **+** symbol between **Start** and **Deny**. A box opens with options for different actions.
4. Click the **Antivirus Check** option button, and then click the **Add Item** button at the bottom of the box.
5. Configure the Properties as applicable for your configuration.
6. Click the **Save** button. You now see two paths, **Successful** and **Fall Back**.
7. Click the **+** symbol on the ***Successful*** path between **Antivirus Check** and **Deny**.
8. Click the **Logon Page** option button, and then click the **Add Item** button at the bottom of the box.
9. Configure the Properties as applicable for your configuration. In our example, we leave the settings at the defaults.
10. Click the **+** symbol on the between **Logon Page** and **Deny**.
11. Click the **SSO Credential Mapping** option button, and then click the **Add** button.
12. Configure the Properties as applicable for your configuration. In our example, we leave the settings at the defaults.
13. Click the **Save** button.
14. On the **SSO Credential Mapping** path, click the **Deny** link box. Click the **Allow** button, and then click **Save**.
15. Click the yellow **Apply Access Policy** link in the upper left part of the window. You must apply an access policy before it takes effect.

16. Click the **Close** button on the upper right to close the VPE.

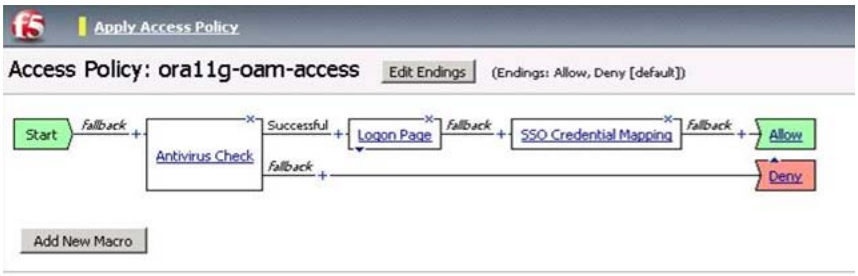

Figure 10 *Final Access Policy in the Visual Policy Editor*

This completes the Access Policy configuration. Continue with the following section to modify the Oracle configuration.

Modifying the Oracle configuration

The primary change that is suggested and which will improve performance can be made to the Oracle configuration. That configuration change is to eliminate the WebGate agent from the application servers if it is present. In the case where the deployment was using WebGate Proxies installed in front of the application, this will not be necessary (the agent was already eliminated to support the separate WebGate Proxy layer).

Oracle states that the Agent typically adds 10-20% degradation to the Web server. If you have previously removed this agent from your servers, you will already have seen this performance improvement on your Web servers.

Reference:
http://download.oracle.com/docs/cd/E12530_01/oam.1014/e10353/deploys.htm#BABDAFFG

Modifying the Oracle Authentication Rule

In this section, we modify the Oracle Authentication rule. This procedure is performed from the Oracle Access Manager Policy Manager console.

To modify the Oracle Authentication rule

1. Browse to your Oracle Access Manager Policy Manager console and log in as an administrator.
2. Navigate to **My Policy Domains** and then click the name of your domain. In our example, we click **IDMEDG**.
3. Click the **Default Rules** tab, and then click the **Authentication Rule** sub-tab.
4. Click the **Actions** link in the menu bar (below the tabs).
5. In the Authentication Success section, for Return complete the following:
   a) In the first **Type** box, type **HeaderVar**.
   b) In the first **Name** box, type **HTTP_OBLIX_UID**.
   c) In the **Return Value** box, type **uid**.
   Move to the set of boxes.
   d) In the second **Type** box, type **HeaderVar**.
   e) In the second **Name** box, type **auth_user**.
   f) In the second **Return Attribute** box, type **uid**.

6. From the Authentication Failure section, in the **Redirection URL** box, type **/my.logout.php3**.
   Note: *This is just the default BIG-IP APM logout page. You could define this to be a custom logout page. See the APM documentation for more information.*
7. In the Authentication Failure section, for Return, complete the following:
   a) In the first **Type** box, type **HeaderVar**.
   b) In the first **Name** box, type **auth_user**.
   c) In the **Return Value** box, type **uid**.
8. Click **Save**.

Figure 11 *Authentication Rule actions*

Enabling OAM and APM integration on the BIG-IP LTM virtual server

The final task in the base configuration is add the Access Profile to the BIG-IP LTM virtual server you created earlier in this guide.

Modifying the virtual server to use the Access Policy

In this procedure, we modify the virtual server you created in *Creating the virtual server*, on page 8 to use the APM Access Profile you created in *Creating an Access Profile*, on page 14.

To modify the virtual server to include the Access Profile

1. On the Main tab, expand **Local Traffic**, and then click **Virtual Servers**. The Virtual Servers screen opens.
2. From the list of virtual servers, click the name of the virtual server you created in *Creating the virtual server*, on page 8. In our example, we click **ora11g-web-vs**.
3. At the bottom of the page in the Access Policy section, from the Access Profile list, select the Access Profile you created in *Creating an Access Profile*, on page 14. In our example, we click **ora11g-oam-access**.
4. Click **Update**.

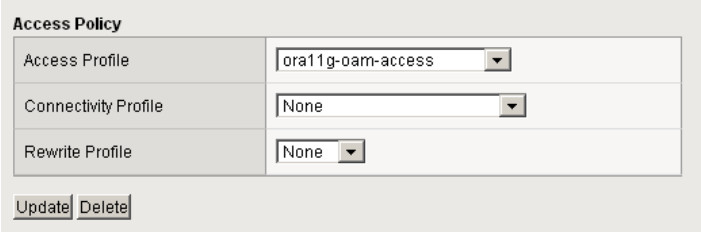

Figure 12 *Access Policy section of the virtual server page*

This completes the base configuration. Now, when an OAM-protected resource is accessed, the BIG-IP APM handles the authentication.

See *Optional: Using an iRule to enable or disable the Access profile*, on page 21 for additional implementation options.

Optional: Using an iRule to enable or disable the Access profile

As mentioned in the prerequisites, we recommend protecting access to ALL of the resources on the backend web servers. However, if you want only portions of the application to require authentication credentials, the next task is to create the iRule.

This iRule enables or disables the Access Profile, depending on whether or not the user is attempting to access a protected resource. This iRule is only necessary if you want to protect resources as defined in your OAM policy. Without this iRule, BIG-IP APM protects all resources behind it.

Note

This iRule is optional, but should be used if you want only portions of the application to require authentication credentials.
We recommend protecting access to all of the resources on the backend web servers, which does not require the use of this iRule.

Creating the iRule

1. On the Main tab, expand **Local Traffic**, and then click **iRules**.
2. Click the **Create** button.
3. In the **Name** box, enter a name for your iRule. In our example, we use **Oracle-SSO-protected-URI-iRule**.
4. In the Definition box, type or paste the following iRule, omitting the line numbers.

```
1    when HTTP_REQUEST {
2            set sid [HTTP::cookie MRHSession]
3
4            if { [ACCESS::session exists $sid] == 1 } {
5                log local0.notice "Valid session found, ACCESS enabled"
6            }
7            else {
8                if { [HTTP::uri] contains "f5iupstate=1" } {
9                    log local0.notice "Protected URI detected, ACCESS enabled"
10               }
11               else {
12                   if { [HTTP::uri] contains "?" } {
13                       set new_uri [HTTP::uri]&f5acmode=1
14                   }
15                   else {
16                       set new_uri [HTTP::uri]?f5acmode=1
17                   }
18
19                   HTTP::uri $new_uri
20                   ACCESS::disable
21               }
22           }
23       }
```

5. Click the **Finished** button.
6. On the Main tab, expand **Local Traffic**, and then click **Virtual Servers**.
7. Click the Virtual Server you created in *Creating the virtual server*, on page 8. In our example, we click **ora11g-web-vs**.
8. On the Menu bar, click **Resources**.
9. In the **iRules** section, click the **Manage** button.
10. From the iRule **Available** list, select the iRule you just created, and then click the Add (**<<**) button.
11. Click **Finished**.

This completes the configuration.