BIG-IP® Access Policy Manager®: OAM Integration Guide

Version 11.2

IT agility. Your way.

Table of Contents

Table of Contents

Legal Notices

Publication Date

This document was published on May 7, 2012.

Publication Number

MAN-0361-02

Copyright

Trademarks

Patents

This product may be protected by U.S. Patent 7,114,180. This list is believed to be current as of May 7, 2012.

Export Regulation Notice

This product may include cryptographic software. Under the Export Administration Act, the United States government may consider it a criminal offense to export this product from the United States.

RF Interference Warning

This is a Class A product. In a domestic environment this product may cause radio interference, in which case the user may be required to take adequate measures.

FCC Compliance

This equipment has been tested and found to comply with the limits for a Class A digital device pursuant to Part 15 of FCC rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This unit generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case the user, at his own expense, will be required to take whatever measures may be required to correct the interference.

Any modifications to this device, unless expressly approved by the manufacturer, can void the user's authority to operate this equipment under part 15 of the FCC rules.

Canadian Regulatory Compliance

This Class A digital apparatus complies with Canadian ICES-003.

Standards Compliance

This product conforms to the IEC, European Union, ANSI/UL and Canadian CSA standards applicable to Information Technology products at the time of manufacture.

Acknowledgments

This product includes software developed by Bill Paul.

This product includes software developed by Jonathan Stone.

This product includes software developed by Manuel Bouyer.

This product includes software developed by Paul Richards.

This product includes software developed by the NetBSD Foundation, Inc. and its contributors.

This product includes software developed by the Politecnico di Torino, and its contributors.

This product includes software developed by the Swedish Institute of Computer Science and its contributors.

This product includes software developed by the University of California, Berkeley and its contributors.

This product includes software developed by the Computer Systems Engineering Group at the Lawrence Berkeley Laboratory.

This product includes software developed by Christopher G. Demetriou for the NetBSD Project.

This product includes software developed by Adam Glass.

This product includes software developed by Christian E. Hopps.

This product includes software developed by Dean Huxley.

This product includes software developed by John Kohl.

This product includes software developed by Paul Kranenburg.

This product includes software developed by Terrence R. Lambert.

This product includes software developed by Philip A. Nelson.

This product includes software developed by Herb Peyerl.

This product includes software developed by Jochen Pohl for the NetBSD Project.

This product includes software developed by Chris Provenzano.

This product includes software developed by Theo de Raadt.

This product includes software developed by David Muir Sharnoff.

This product includes software developed by SigmaSoft, Th. Lockert.

This product includes software developed for the NetBSD Project by Jason R. Thorpe.

This product includes software developed by Jason R. Thorpe for And Communications, http://www.and.com.

This product includes software developed for the NetBSD Project by Frank Van der Linden.

This product includes software developed for the NetBSD Project by John M. Vinopal.

This product includes software developed by Christos Zoulas.

This product includes software developed by the University of Vermont and State Agricultural College and Garrett A. Wollman.

This product includes software developed by Balazs Scheidler (bazsi@balabit.hu), which is protected under the GNU Public License.

This product includes software developed by Niels Mueller (nisse@lysator.liu.se), which is protected under the GNU Public License.

In the following statement, *This software* refers to the Mitsumi CD-ROM driver: This software was developed by Holger Veit and Brian Moore for use with 386BSD and similar operating systems. *Similar operating systems* includes mainly non-profit oriented systems for research and education, including but not restricted to NetBSD, FreeBSD, Mach (by CMU).

This product includes software developed by the Apache Group for use in the Apache HTTP server project (http://www.apache.org/).

This product includes software licensed from Richard H. Porter under the GNU Library General Public License (© 1998, Red Hat Software), www.gnu.org/copyleft/lgpl.html.

This product includes the standard version of Perl software licensed under the Perl Artistic License (© 1997, 1998 Tom Christiansen and Nathan Torkington). All rights reserved. You may find the most current standard version of Perl at http://www.perl.com.

This product includes software developed by Jared Minch.

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/).

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

This product contains software based on oprofile, which is protected under the GNU Public License.

This product includes RRDtool software developed by Tobi Oetiker (http://www.rrdtool.com/index.html) and licensed under the GNU General Public License.

This product contains software licensed from Dr. Brian Gladman under the GNU General Public License (GPL).

This product includes software developed by the Apache Software Foundation (http://www.apache.org/).

This product includes Hypersonic SQL.

This product contains software developed by the Regents of the University of California, Sun Microsystems, Inc., Scriptics Corporation, and others.

This product includes software developed by the Internet Software Consortium.

This product includes software developed by Nominum, Inc. (http://www.nominum.com).

This product contains software developed by Broadcom Corporation, which is protected under the GNU Public License.

This product contains software developed by MaxMind LLC, and is protected under the GNU Lesser General Public License, as published by the Free Software Foundation.

Chapter 1

Integrating Access Policy Manager with OAM

Topics:

About integration with supported Oracle Access Manager versions

Access Policy Manager® can provide the same functionality as an Oracle 10g WebGate. Access Policy Manager native OAM integration is built on top of Oracle® 10g's latest Access Manager SDK. When you deploy Access Policy Manager with an OAM 10g or 11g server and OAM 10g WebGates, you no longer need to deploy a WebGate proxy or WebGate agent for each OAM-protected web application.

Access Policy Manager supports multiple WebGates and can function as an Authentication WebGate (when deployed with Oracle 10g server) as well as a Resource WebGate (when deployed with either Oracle 10g or 11g server).

Authentication WebGate (AWG)	The front-end agent of the OAM server that provides the interface of authentication and authorization for the user's access request to specific web resources.
Resource WebGate (RWG)	The front-end agent of protected web servers; the RWG validates the OAM session cookie (ObSSOCookie) to determine whether the user has been authenticated and can be authorized to access the requested web resources.

Although the Oracle 11g server is backward compatible with Oracle 10g WebGates, with Oracle 11g, Access Policy Manager acts in place of OAM 10g resource webgates, but cannot act as a authentication webgate. This is because a new architecture was introduced with OAM 11g in which the OAM 11g server becomes the central management point for everything including authentication, that is, the role of AWG. Refer to *Oracle® Fusion Middleware Administrator's Guide for Oracle Access Manager 11g* for a comparison of OAM 10g and 11g architectures.

Because the Oracle 11g server handles all user authentication requests, you should take steps to prevent and mitigate Layer 7 Denial of Server (DoS) and brute force attacks by installing a Web Application Firewall in front of the Oracle 11g server. BIG-IP® Application Security Manager® can provide you with intelligent Layer 7 protection in this case. For more information, refer to *Configuration Guide for BIG-IP® Application Security Manager®*.

How does native integration with OAM work?

You can achieve SSO functionality with OAM for HTTP/HTTPS requests passing through a virtual server to the web application. With OAM support enabled on a Local Traffic Manager® (LTM) virtual server, Access Policy Manager® will be the OAM policy enforcement point (PEP) on the BIG-IP® system, while the OAM server is still the policy decision point (PDP) in the overall system architecture. When a user requests access to a protected web resource, Access Policy Manager® communicates with the OAM server to determine whether the user can be authenticated/authorized for the request, and enforces the policy evaluation decision (made by OAM server) on the BIG-IP® device.

The figures that follow show a typical configuration before and after OAM native integration is enabled.

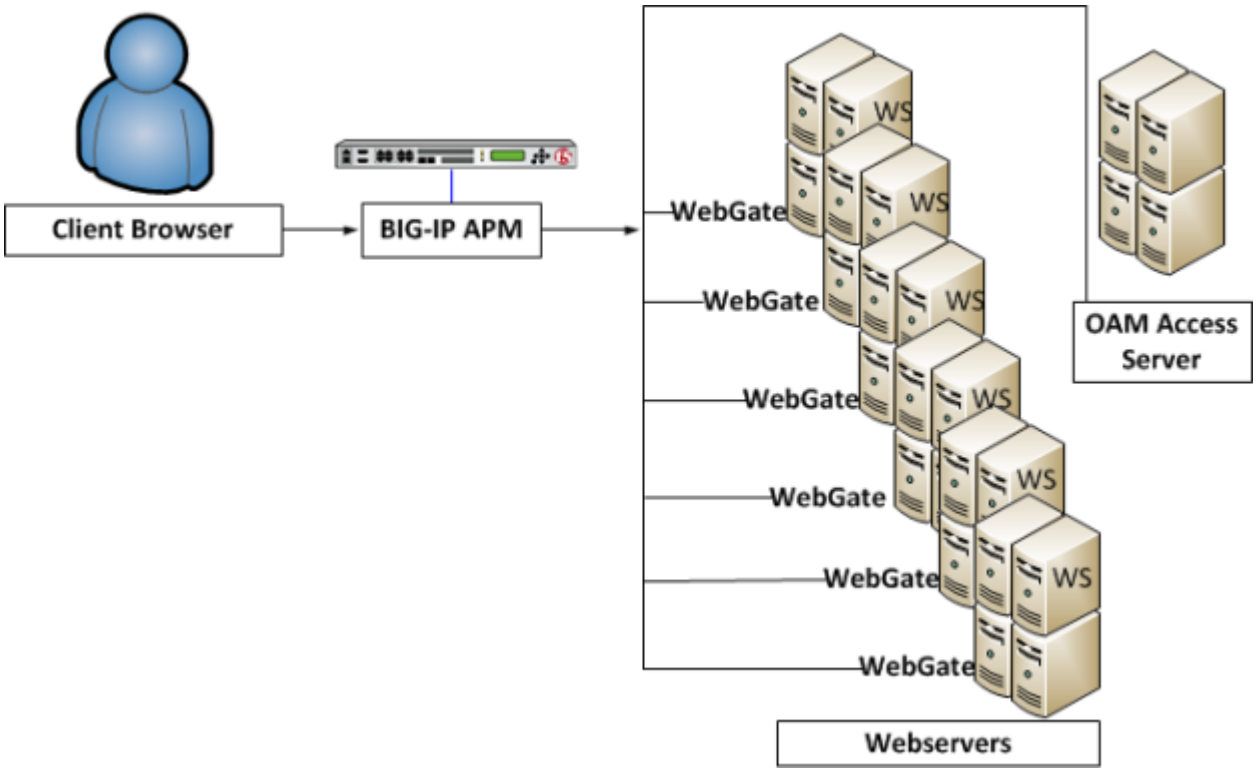

Figure 1: Typical configuration before OAM native integration is enabled on the BIG-IP system

In this figure individual WebGates, installed on each webserver, interact with the OAM Access Server.

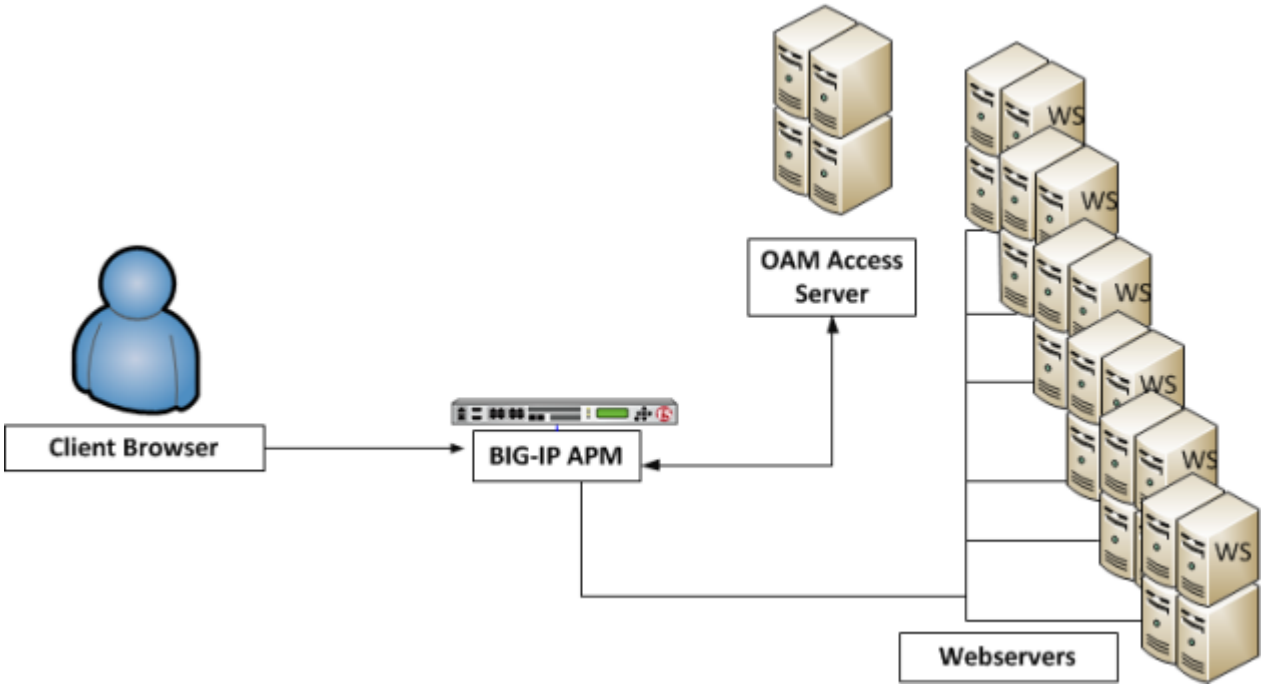

Figure 2: Typical configuration after OAM native integration is enabled on the BIG-IP system

In this figure WebGates are no longer required on the webservers, and, even if they are installed, they are not used. Access Policy Manager acts in place of the WebGates, contacting the OAM Access Server for policy information, and enforcing the policies.

OAM 11g SSO integration example

Let's walk through an example deployment with Oracle 11g. You can integrate Access Policy Manager® with a Oracle 11g server whether it is configured for single sign on (SSO) single domain or SSO multi-domain. To keep this example simple, we will assume that Oracle 11g server is configured for SSO single domain. The Oracle 11g server performs all authentication. A single Resource WebGate is configured in OAM.

In Access Policy Manager on the BIG-IP® system, a AAA OAM server has been configured and includes the details of the OAM Access Server and one AccessGate. One virtual server has been configured with OAM native integration enabled. BIG-IP® Application Security Manager® (ASM) is installed in another virtual server as a web application firewall configured to prevent DoS and mitigate brute force attacks.

This figure depicts the traffic flow for the example.

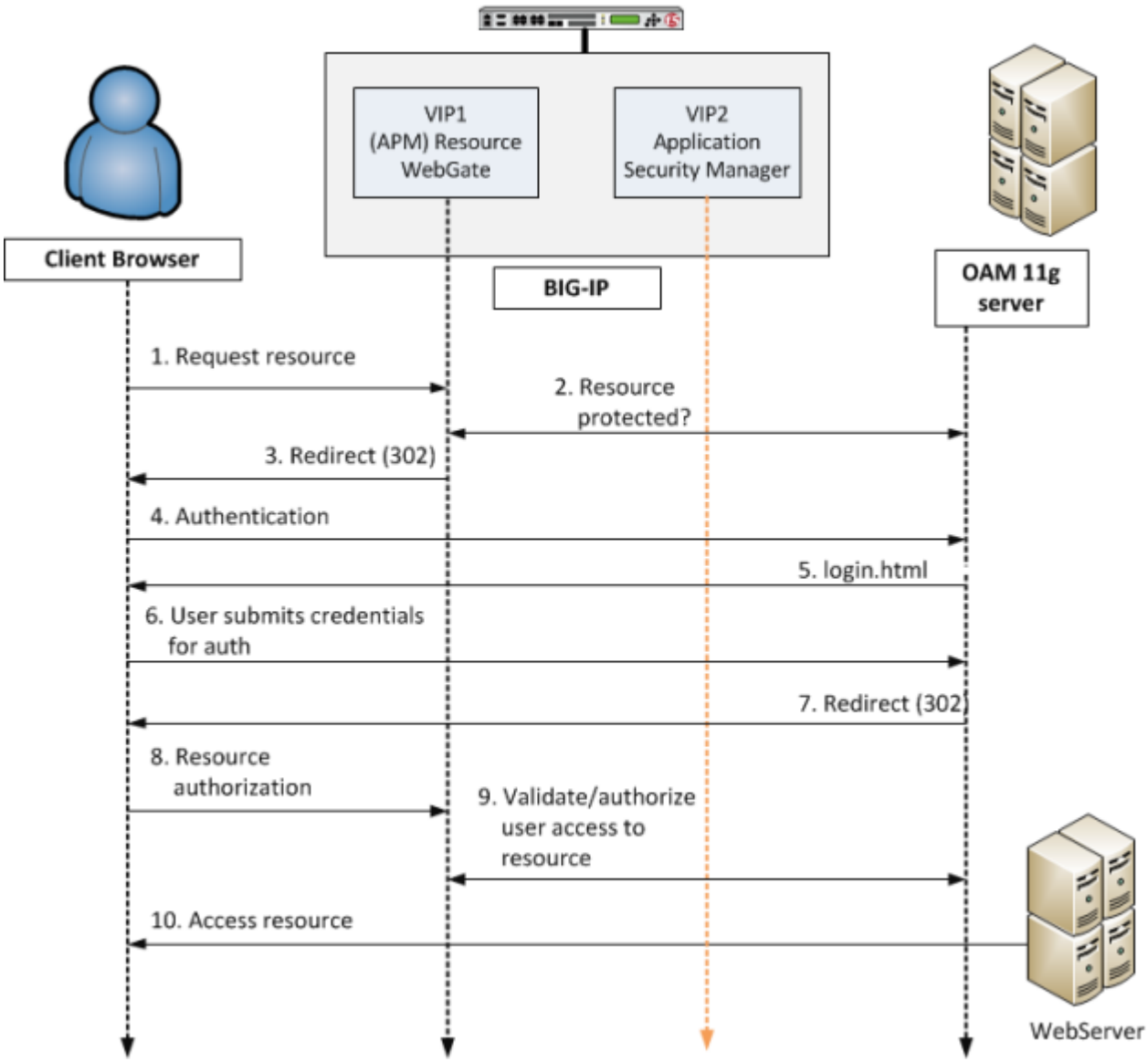

Figure 3: Accessing a protected resource using Access Policy Manager deployed with OAM 11g

1. Client requests access to a resource. The request comes to the Resource Webgate (RWG).
2. RWG checks whether the resource is protected per OAM. The resource is protected and the user has not yet authenticated.
3. RWG sends a 302 redirect to the client so that the client will be redirected to the OAM 11g server for authentication.
4. User will follow the redirect to OAM 11g server for authentication. In this example, the user has never been authenticated and form-based authentication is the authentication scheme of the OAM policy protecting the original user-requested resource.

Note: *Before going to OAM, traffic is checked against security policies that are configured with anomaly protection on ASM, provided that the ASM module is enabled to protect the OAM 11g server on the BIG-IP system.*

5. OAM sends a login page to the client.
6. User submits credentials which come to OAM server where the user's credentials will be validated. In this example, it is assumed that the user submitted valid credentials.
7. After user credentials are successfully validated on the OAM 11g server, the server will send another 302 redirect, so that the user will be redirected back to the original RWG.
8. Resource request comes to RWG.
9. RWG verifies the user's original request again using the `ObSSOCookie` passed from the OAM 11g server. Upon successful authorization, the user will be allowed to access the resource.
10. The protected resource behind VIP1 will be sent back to the user.

OAM 10g SSO integration example

Let's walk through an example deployment. An Oracle 10g server is configured for SSO multi-domain; an Authentication WebGate is configured and, in another domain, a Resource WebGate is configured.

In Access Policy Manager®, an AAA OAM server has been configured and includes the details of the OAM Access Server and the two AccessGates. Two virtual servers have been configured with OAM native integration enabled.

This figure depicts the traffic flow for the example.

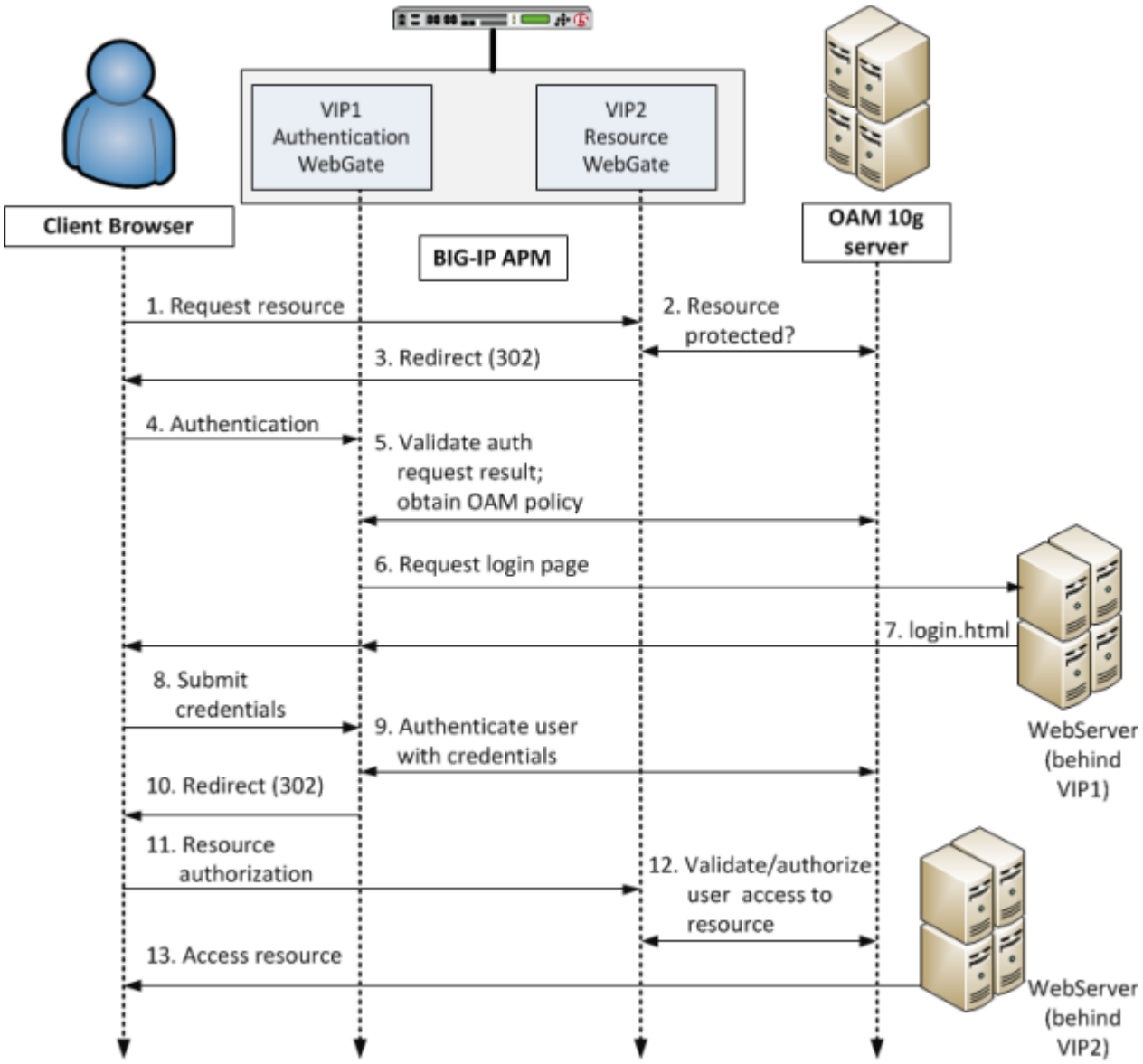

Figure 4: Accessing a protected resource via Access Policy Manager native integration with OAM 10g

1. Client requests access to a resource. The request comes to the RWG (Access Policy Manager AccessGate at VIP2).
2. RWG checks whether the resource is protected per OAM. The resource is protected and the user has not yet authenticated.
3. RWG sends a 302 redirect to the client so that the client will be redirected to the AWG for authentication.
4. Authentication request comes to the AWG (Access Policy Manager AccessGate at VIP1).
5. AWG validates user authentication status with OAM and obtains policy. In this case, the policy calls for form-based authentication and gives the location of the form.
6. For the form-based authentication scheme, AWG allows the user to access the login page hosted on a webserver behind the AWG.
7. The webserver returns the login.html file to the AWG, which sends it to the client.
8. Via login.html, the user submits credentials.

9. The AWG uses the credentials to authenticate the user with the OAM 10g server.
10. With user authentication successful, the AWG sends a 302 redirect to the client so that the client will be redirected to the original RWG.
11. Request for resource comes to the RWG again.
12. The RWG validates user access to the resource with OAM.
13. The protected resource behind VIP2 will be sent back to the user.

Chapter 2

Access Policy Integration

Topics:

About AAA OAM server configuration

You can configure only one AAA OAM server, but it can support multiple AccessGates from the same access server. When you create a AAA OAM server, its transport security mode must match the setting in the OAM access server.

Task summary for integrating Access Policy Manager with OAM

Before you begin

Before you start to integrate Access Policy Manager® with OAM, configure the Access Server and AccessGates through the Oracle Access administrative user interface. Refer to *Oracle® Access Manager Access Administration Guide* for steps.

Task list

Follow these steps to integrate Access Policy Manager with a supported OAM server.

Importing AccessGate files when transport security is set to cert
Creating an AAA OAM server
Adding AccessGates to the OAM AAA server
Creating a virtual server

Importing AccessGate files when transport security is set to cert

Check the transport security mode that is configured on the OAM access server. If transport security mode is configured to cert, copy the certificate,certificate chain, and key files (by default, `aaa_cert.pem`, `aaa_chain.pem`, and `aaa_key.pem` respectively) for each AccessGate from the OAM access server to the BIG-IP system.

Note: *If Transport Security Mode is set to open or simple, you can skip this procedure.*

You must import the certificate, certificate chain, and key files for each AccessGate into the BIG-IP system. Repeat this procedure for each AccessGate. Import certificate and certificate chain files before importing the corresponding private key file.

Note: *If a signing chain certificate (CA) is the subordinate of another Certificate Authority, both certificates, in PEM format, must be included in the file with the subordinate signer CA first, followed by the root CA, including " -----BEGIN/END CERTIFICATE-----".*

1. On the Main tab, click **Local Traffic** > **SSL Certificate List**.
   The SSL Certificate List screen opens.
2. In the **Import Type** list, select **Certificate**.
3. For the **Certificate Name** setting, select the **Create New** option, and type a unique name that enables you to identify the file as belonging to this particular AccessGate.

4. For the **Certificate Source** setting, select the **Upload File** option, and browse to the location of the certificate or the certificate chain file.

   If you kept the default filenames when you copied the files to the BIG-IP system, look for `aaa_cert.pem` or `aaa_chain.pem`.

5. Click **Import**.
   A certificate or certificate chain file has been imported for the AccessGate. To import the other (certificate or certificate chain) file for this AccessGate, repeat the steps that you have just completed before you continue.
6. On the Main tab, click **Local Traffic** > **SSL Certificate List**.
   The SSL Certificate List screen opens.
7. From the **Import Type** list, select **Key**.
8. For the **Key Name** setting, select the **Create New** option, and type a unique name that enables you to identify the file as belonging to this particular AccessGate.

   When you import the key file, you are importing the private key that corresponds to the already imported certificate and certificate chain while renaming the file from its default name `aaa_key.pem`.

9. For the **Key Source** setting, do one of the following:

   - Select the **Upload File** option, and browse to the location of the key file.
   - Select the **Paste Text** option, and paste the key text copied from another source.

10. Click **Import**.
    The key file is imported.

Certificate, certificate chain, and key files have been imported for an AccessGate.

Repeat the procedure to import these files for any other AccessGate.

Creating an AAA OAM server

If transport security mode is configured to cert on the access server, import the certificates, keys, and CA certificate for the AccessGates into the BIG-IP system.

Create a AAA server for OAM to deploy Access Policy Manager® in place of OAM 10g WebGates.

Note: *Only one OAM server per BIG-IP system is supported. Multiple OAM 10g webgates from the same OAM server are supported.*

1. In the navigation pane, click **Access Policy** > **AAA Servers** > **Oracle Access Manager**.
   The Oracle Access Manager Server screen opens.
2. Click **Create** if no Oracle Access Manager server is defined yet,.
   The New OAM Server screen opens.
3. Type a name for the AAA OAM server.
4. For **Access Server Name**, type the name that was configured in Oracle Access System for the access server.

   For the access server name, open the OAM Access System Console and select **Access system configuration** > **Access Server Configuration**.

5. For **Access Server Hostname**, type the fully qualified DNS host name for the access server system.
6. For **Access Server Port**, accept the default 6021, or type the port number.
7. For **Admin Id**, type the admin ID.

Admin Id and Admin Password are the credentials that are used to retrieve host identifier information from OAM. Usually, these are the credentials for the administrator account of both Oracle Access Manager and Oracle Identity Manager.

8. For **Admin Password**, type the admin password.
9. For **Retry Count**, accept the default 0, or enter the number of times an AccessGate should attempt to contact the access server.
10. For **Transport Security Mode**, select the mode (open, simple, or cert) that is configured for the access server in Oracle Access System.
11. If Transport Security Mode is set to simple, type and re-type a **Global Access Protocol Passphrase**; it must match the global passphrase that is configured for the access server in OAM.
12. For **AccessGate Name**, type the name of an AccessGate; it must match the name of an AccessGate that is configured on the OAM access server.
13. For **AccessGate Password** and **Verify Password**, type the password; it must match the password that is configured for it on the OAM access server.
14. If transport security mode is set to cert, select the **Certificate**, **Key**, and **CA Certificate** that you imported for this particular AccessGate.
15. If transport security mode is set to cert and if a sign key passphrase is needed, type a **Sign Key Passphrase** and re-type it to verify it.
16. Click the **Finished** button.
    This adds the new AAA server to the AAA Servers list.

Add any other AccessGates that are configured for the OAM access server to this Oracle Access Manager AAA server. Then, for each AccessGate, configure a virtual server and enable OAM support on it for native integration with OAM.

Adding AccessGates to the OAM AAA server

You must create an Oracle Access Manager AAA server with one AccessGate before you can add other AccessGates.

Access Policy Manager can support multiple AccessGates from the same OAM access server. To enable the support, add the AccessGates to the Oracle Access Manager AAA server.

1. In the navigation pane, click **Access Policy** > **AAA Servers** > **Oracle Access Manager**.
   The Oracle Access Manager Server screen opens.
2. Click the name of the Oracle Access Manager AAA server.
   The Properties page opens.
3. Scroll down to the **AccessGate List** and click **Add**.
   The New AccessGate page opens.
4. For **AccessGate Name**, type the name of an AccessGate; it must match the name of an AccessGate that is configured on the OAM access server.
5. For **AccessGate Password** and **Verify Password**, type the password; it must match the password that is configured for it on the OAM access server.
6. If transport security mode is set to cert for the access server, select the **Certificate**,**Key**, and **CA Certificate** that you imported for this particular AccessGate.
7. If transport security mode is set to cert for the access server, and if a sign key passphrase is needed, type a **Sign Key Passphrase** and re-type it to verify it.
8. Click the **Finished** button.

The AccessGate is added.

Creating a virtual server

Configure an AAA OAM server and add AccessGates to it before you perform this task.

A virtual server represents a destination IP address for application traffic. Configure one virtual server for each AccessGate that is included on the AAA OAM server AccessGates list.

1. On the Main tab, click **Local Traffic** > **Virtual Servers**.
   The Virtual Server List screen displays a list of existing virtual servers.
2. Click the **Create** button.
   The New Virtual Server screen opens.
3. For the **Destination** setting, in the **Address** field, type the IP address you want to use for the virtual server.

   The IP address you type must be available and not in the loopback network.
4. In the **Service Port** field, type a port number or select a service name from the **Service Port** list.
5. In the Resources area of the screen, from the **Default Pool** list, select a pool name.
6. Scroll down to the Access Policy section and check the **Enabled** box for OAM Support.
7. Select an AccessGate from the list.

   If you select `Default`, Access Policy Manager reads Oracle configuration information to determine which AccessGate to associate with this virtual server.
8. Click **Finished**.

A destination IP address on the Access Policy Manager® system is now available for application traffic.

Troubleshooting tips

You might run into problems with the integration of Access Policy Manager® and OAM in some instances. Follow these tips to try to resolve any issues you might encounter.

Troubleshooting tips for initial configuration

You should	Steps to take
Check network connectivity	Ping the OAM Access Server from the BIG-IP system.
Test without OAM support enabled first	Before you test with OAM support enabled, make sure that the BIG-IP system has basic connectivity to protected applications. • Disable the OAM Support property on the virtual server. • Verify that you can reach the pool and the application. After succeeding, reenable OAM support on the virtual server.
Check the configuration for accuracy	• Confirm that the AAA server object is correct, particularly the OAM server section. • Confirm that the AccessGates configured on the BIG-IP system within the AAA server are correct.

Additional troubleshooting tips

You should	Steps to take
Verify access	OAM provides tools for the administrator to test how access policies respond to various requests. Use the Access Tester to test access policies with given identities and for given users. This tool can be helpful in determining whether the access provided by BIG-IP system is consistent with the policies configured under OAM.
Resolve sudden problems	Changes that have been made on the OAM server can cause mismatches on the BIG-IP system due to a configuration cache that is kept on the BIG-IP system. To resolve this problem, clear the cache on the BIG-IP system. • Select **Access Policy > AAA Servers > Oracle Access Manager** and click the name of AAA server configuration. • In the Access Server Configuration section, click the **Clear Local Config Cache** button.
Check logs	Enable and review the log files on the BIG-IP system. • Most relevant log items are kept in the /var/log/apm log file. This /var/log/apm log file is the primary location for messages related to the operation of OAM. • Additional logging is done in /var/log/oblog.log. This file contains AccessGate logging which might be helpful in certain circumstances.

Using OAM authentication in an access policy

Before you start this procedure, Access Server and AccessGates must be configured through the Oracle Access administrative user interface. An Access Policy Manager® AAA OAM server and a virtual server must be configured on the BIG-IP® system.

Configure OAM authentication in an access policy only if you need to provide a client with SSL VPN access, authenticating with an Oracle server that is configured for single sign on single domain use. This approach does not work for Oracle single sign on multi-domain configurations.

Note: *You do not need an access policy to use Access Policy Manager as an OAM 10g Webgate.*

Tip: *In this procedure, you create a new access profile as part of the configuration. Alternatively, you can edit an existing access profile and add OAM authentication to the access policy.*

1. On the Main tab, click **Access Policy** > **Access Profiles**.
   The Access Profiles List screen opens.
2. Click **Create**.
   The New Profile screen opens.
3. Type a name for the access profile.
4. Click **Finished**.
5. Click the name of the access profile for which you want to edit the access policy.

The Access Profile properties screen opens for the profile you want to edit.

6. Click **Edit Access Policy for Profile *profile_name***.
   The visual policy editor opens the access policy in a separate window or tab.
7. Click the **[+]** sign anywhere in your access profile to add your new policy action item.
   An Add Item window opens, listing Predefined Actions that are grouped by General Purpose, Authentication, and so on.
8. Select **OAM**, and click **Add item**.
9. For Server, select the AAA OAM server from the list.
10. For URL, type in a URL resource.
11. For Agent Action, select either Authentication and Authorization or Authentication Only.
12. Click **Save**.
    You will return to the visual policy editor.
13. Click **Apply Access Policy** to save your configuration.

The access policy associated with the AAA OAM server uses OAM authentication.

Index