F5 BIG-IP Access Policy Management Operations Guide 1.0

Comprehensive Global Access Anytime, Anywhere

With BIG-IP Access Policy Manager (APM), your network, cloud, and applications are secure. BIG-IP APM provides valuable insight into who is on your network or cloud, which applications they're accessing, with which devices, from where, and when.

CONTENTS

List of tables

List of figures

MANAGEMENT

7.1 Variable Assign access policy agent used to collect license usage
7.2 Branch rules in Variable Assign agent collect license usage information in session variables
7.3 Logout page error message configuration
7.4 Logout page example as seen by user

ACCESS PROGRAMMABILITY

8.1 Access iRules event diagram
8.2 All Sessions tab in session reports interface
8.3 Session Variables report tab
8.4 Session variables displayed using -allkeys command in sessiondump
8.5 Access policy Logging agent Properties tab configuration
8.6 Session variable information in BIG-IP APM log messages
8.7 Access policy Message Box agent Properties tab configuration
8.8 Message Box as seen by user
8.9 Access policy Variable Assignment agent Custom Variable configuration
8.10 Access policy Variable Assignment agent Custom Expression configuration
8.11 Access policy Logon Page agent Secure Custom Variable configuration
8.12 Access policy SSO Credential Mapping agent Unsecure Custom Variable configuration

TROUBLESHOOTING

9.1 Access policy using Logon Page and AD Auth policy agents.

Legal notices

Publication date

This document was published on May 8, 2015.

Publication Number: BIG-IP APMOps 01_0.

Copyright

Trademarks

AAM, Access Policy Manager, Advanced Client Authentication, Advanced Firewall Manager, Advanced Routing, AFM, APM, Application Acceleration Manager, Application Security Manager, ARX, AskF5, ASM, BIG-IP, BIG-IQ, Cloud Extender, CloudFucious, Cloud Manager, Clustered Multiprocessing, CMP, COHESION, Data Manager, DevCentral, DevCentral [DESIGN], DNS Express, DSC, DSI, Edge Client, Edge Gateway, EdgePortal, ELEVATE, EM, Enterprise Manager, ENGAGE, F5, F5[DESIGN], F5 Certified [DESIGN], F5 Networks, F5SalesXchange [DESIGN], F5Synthesis, f5Synthesis, F5Synthesis[DESIGN], F5 TechXchange [DESIGN], Fast Application Proxy, Fast Cache, FirePass, Global Traffic Manager, GTM, GUARDIAN, iApps, IBR, Intelligent Browser Referencing, Intelligent Compression, IPv6 Gateway, iControl, iHealth, iQuery, iRules, iRules OnDemand, iSession, L7 RateShaping, LC, Link Controller, Local Traffic Manager, LTM, LineRate, LineRate Systems [DESIGN], LROS, LTM, Message Security Manager, MSM, OneConnect, Packet Velocity, PEM, Policy Enforcement Manager, Protocol Security Manager, PSM, Real Traffic Policy Builder, SalesXchange, ScaleN, Signalling Delivery Controller, SDC, SSL Acceleration, software designed applications services, SDAC (except in Japan), StrongBox, SuperVIP, SYN Check, TCP Express, TDR, TechXchange, TMOS, TotALL, Traffic Management Operating System, Traffix Systems, Traffix Systems (DESIGN), Transparent Data Reduction, UNITY, VAULT, vCMP, VE F5 [DESIGN], Versafe, Versafe [DESIGN], VIPRION, Virtual Clustered Multiprocessing, WebSafe, and ZoneRunner, are trademarks or service marks of F5 Networks, Inc., in the U.S. and other countries, and may not be used without express written consent.

All other product and company names herein may be trademarks of their respective owners.

Patents

This product may be protected by one or more patents. See the ***F5 Patents*** page (http://www.f5.com/about/guidelines-policies/patents).

Notice

THE SOFTWARE, SCRIPTING, AND COMMAND EXAMPLES ARE PROVIDED "AS IS," WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES, OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE, SCRIPTING AND COMMAND EXAMPLES, OR THE USE OR OTHER DEALINGS WITH THE SOFTWARE, SCRIPTING, AND COMMAND EXAMPLES.

Acknowledgements

Executive sponsor: Julian Eames, Executive Vice President, Business Operations

Publisher and project manager: Jeanne Lewis

Editor: Andy Koopmans

Project team, writers, editors, and testers: John Harrington, Maxim Ivanitskiy, Amy Knight, Vladimir Kokshenev, Bipin Kumar, Nishant Kumar, Jatin Parmar, Dan Pruett, Svetlana Rudyak, Rick Salsa, Kevin Stewart, Lucas Thompson, Alexey Vasilyev, and A. Lee Wade.

BookSprints facilitators, designer, editor, and support team: Laia Ros, Barbara Rühling, Henrik van Leeuwen, Julien Taquet, Raewyn Whyte, and Juan Gutiérrez. For more information on the BookSprints proccess, see the ***BookSprints*** web site. (This link takes you to an outside resource.)

Content, support, and assistance: Don Martin, Vice President, Global Services New Product & Business Development; the Global Services New Product Introduction Team, Bryan Gomes, Phillip Esparza, Derek Smithwick, Beth Naczkowski, Joe Taylor, Mark Kramer, Andrew Pemble, Dave Bowman, Jim Williams, David Katz; and the rest of the Global Services management team. Thanks also to the BIG-IP APM product development team, Walter Griffeth, James Goodwin, Satoshi Asami, Ravi Natarajan and Piyush Jain; Joe Scherer, Regional Vice President, Field Systems Engineering; and Ignacio Avellaneda, Colin Hayes, and Marian Salazar.

About this guide

This guide includes recommended maintenance and monitoring procedures related to F5® BIG-IP® Access Policy Manager® (APM) versions 11.2.1–11.6.0.

The goal of this guide is to assist F5 customers with keeping the BIG-IP APM system healthy, optimized, and performing as designed. It was written by F5 engineers who assist customers with solving complex problems every day. Some of these engineers were customers before joining F5. Their unique perspective and hands-on experience has been leveraged in this guide to to serve the operational and maintenance guides F5 customers have requested.

This guide describes common information technology procedures and some that are exclusive to BIG-IP systems. There may be procedures particular to your industry or business that are not identified. While F5 recommends the procedures outlined in this guide, they are intended to supplement your existing operations requirements and industry standards. F5 suggests that you read and consider the information provided to find the procedures to suit your implementation, change management process, and business operations requirements. Doing so can result in fewer unscheduled interruptions and higher productivity.

See *Feedback* at the end of this section for information about how to help improve future versions of the guide.

Before using this guide

You will get the most out in this guide if you have already completed the following, as appropriate to your implementation:

- Installed your F5 platform according to its requirements and recommendations. Search the ***AskF5 Knowledge Base*** (support.f5.com) for "platform guide" to find the appropriate guide.
- Followed the general environmental guidelines in the hardware platform guide to make sure of proper placement, airflow, and cooling.
- Set recommended operating thresholds for your industry, accounting for seasonal changes in load. For assistance, you can contact ***F5 Consulting Services***.

- Familiarized yourself with F5 technology concepts and reviewed and applied appropriate recommendations from ***F5 BIG-IP TMOS: Operations Guide*** and ***F5 BIG-IP: Local Traffic Manager and Glocal Traffic Manager Operations Guide*****.**

Limits of this guide

This guide does not address installation, setup, or configuration of your BIG-IP system or modules.

There is a wealth of documentation covering these areas in ***AskF5 Knowledge Base*** (support.f5.com) The F5 self-help community, ***DevCentral*** (devcentral.f5.com), is also a good place to find answers about initial deployment and configuration. You can find additional resources detailed in the *Optimize the Support Experience* chapter of this guide.

The following figure shows where this guide can best be applied in the product life cycle.

Figure 0.1: BIG-IP APM documentation coverage

Glossary

A glossary is not included in this document. Instead, the ***Glossary and Terms*** page (www.f5.com/glossary) offers an up-to-date and complete listing and explanation of common industry and F5-specific terms.

Customization

Customizing BIG-IP APM may benefit your implementation. You can get help with customization from a subject matter expert, such as a professional services consultant from ***F5 Consulting Services*** (f5.com/support/professional-services).

Issue escalation

See *Optimize the Support Experience* this guide for escalation guidance. Customers with websupport contracts can also open a support case by clicking **Open a support case** on the ***AskF5 Knowledge Base*** page (support.f5.com)

Feedback and notifications

F5 welcomes feedback and requests and invites you to visit our ***F5 Operations Guide User Feedback survey*** (This link sends you to an external site.)

F5 operations guides are updated frequently and new guides are being written. If you would like to be notified when new content is available, email **opsguide@f5.com** and your name will be added to our distribution list for updates and new releases.

Document conventions

To help you easily identify and understand important information, the document in this guide uses the stylistic conventions described here.

Examples

All examples in this document use only private IP addresses. When you set up the configurations described, you will need to use valid IP addresses suitable to your own network in place of our sample addresses.

References to objects, names, and commands

We apply **bold text** to a variety of items to help you easily pick them out of a block of text. These items include interface labels, specific web addresses, IP addresses, utility names, and portions of commands, such as variables and keywords. For example, with the **tmsh** list self <**name**> command, you can specify a specific self-IP address to show by specifying a name for the <**name**> variable.

References to other documents

We use *italic text* to denote a reference to a chapter or section in this guide or another document. We use ***bold, italic text*** to denote a reference to another document or internet page. For example, for installation instructions see *Performing the Installation* in ***BIG-IP Systems: Getting Started Guide***.

Note Unless otherwise noted, all documents referenced in this guide in bold italic style can be found by searching by title at ***AskF5*** (support.F5.com).

Configuration utility

The BIG-IP® **Configuration utility** is the name of the graphic user interface (GUI) of the BIG-IP system and its modules. It is a browser-based application you can use to install, configure, and monitor your BIG-IP system.

Configuration utility menus, submenus, links, and buttons are formatted in bold text. For more information about the **Configuration utility**, see *Introducing BIG-IP Systems* in ***BIG-IP Systems: Getting Started Guide***.

Command line syntax

We show command line input and output in courier font.

The corresponding prompt is not included. For example, the following command shows the configuration of the specified pool name:

```
tmsh show /ltm pool my_pool
```

The following table explains additional special conventions used in command line syntax:

Table 0.1: Command-line syntax conventions

Character	Description
<>	Identifies a user-defined variable parameter. For example, if the command has <your name>, type in *your name* but do not include the brackets.
[]	Indicates that syntax inside the brackets is optional.
...	Indicates that you can type a series of items.

TMOS shell syntax

The BIG-IP system includes a tool known as the **TMOS shell (tmsh)** that you can use to configure and manage the system from the command line. Using tmsh, you can configure system features, and set up network elements. You can also configure the BIG-IP system to manage local and global traffic passing through the system, and view statistics and system performance data.

You can run **tmsh** and issue commands in the following ways:

- You can issue a single **tmsh** command at the BIG-IP system prompt using the following syntax:

```
tmsh [command] [module . . . module] [component] (options)
```

- You can open tmsh by typing **tmsh** at the BIG-IP system prompt:

```
(tmos)#
```

  Once at the tmos prompt, you can issue the same command syntax, leaving off **tmsh** at the beginning.

  For the sake of brevity all tmsh commands provided in this guide appear in the first format.

Note You can use the command line utilities directly on the BIG-IP system console, or you can run commands using a remote shell, such as the SSH client or a Telnet client. For more information about command line utilities, see ***Bigpipe Utility Reference Guide*** or the ***Traffic Management Shell (tmsh) Reference Guide***.

Introduction

BIG-IP APM features

Client-side interaction

BIG-IP APM with other BIG-IP modules

BIG-IP APM features

BIG-IP APM is a software module of the BIG-IP hardware platform that provides users with secured connections to BIG-IP Local Traffic Manager™ (LTM) virtual servers, specific web applications, or the entire corporate network.

BIG-IP APM is built around several features including access profiles, access policies, the Visual Policy Editor, and webtops.

For more introductory information about BIG-IP APM, see ***BIG-IP APM Documentation***

Access profile

An access profile is the profile you select in a BIG-IP LTM virtual server definition to establish a secure connection to a resource, such as an application or a webtop. Access profiles can be configured to provide access control and security features to a local traffic virtual server hosting web applications.

An access profile contains the following:

- Access session settings.
- Access policy timeout and concurrent user settings.
- Accepted and default language settings.
- Single sign-on (SSO) information and cookie parameter settings.
- Customization settings.
- The access policy for the profile.

For more information, see ***Creating Access Profiles and Access Policies*** in ***BIG-IP Access Policy Manager: Network Access*** and ***Customizing Access Policy Manager Features*** in ***BIG-IP Access Policy Manager: Customization***.

Access policy

An access policy is an object where you define criteria for granting access to various servers, applications, and other resources on your network.

A policy may contain the following:

- One start point
- One or more actions
- Branches
- Macros or macro calls
- One or more endings

An access policy allows you to perform four basic tasks:

- Collect information about the client system.
- Use authentication to verify client security against external authentication servers.
- Retrieve a user's rights and attributes.
- Grant access to resources.

For more information, see ***Creating an Access Policy*** in ***BIG-IP Access Policy Manager: Network Access***.

Visual Policy Editor

The Visual Policy Editor (VPE) is a tool within BIG-IP APM **Configuration utility** for configuring access policies using visual elements.

The elements used to build an access policy in the VPE are called by various names in F5 documentation. In this guide, they are referred to as policy "agents." For example, the **AD Auth** policy agent or **AD Auth** agent.

For more information on VPE conventions, see ***Visual Policy Editor*** in ***BIG-IP Access Policy Manager: Visual Policy Editor.***

Webtop

A webtop is a landing page through which resources are made available to users. There are three types of webtops you can configure:

- A **network access webtop** provides a landing page for an access policy branch to which you assign only a network resource.

- A **portal access webtop** provides a landing page for an access policy branch to which you assign only portal access resources.
- A **full webtop** provides an access policy ending for a branch to which you can assign portal access resources, app tunnels, remote desktops, and/or webtop links, in addition to a network access tunnel.

For more information, see ***Configuring webtops*** in ***BIG-IP Access Policy Manager: Network Access.***

Client interaction with BIG-IP APM

Understanding the basic protocol flow between a client and BIG-IP APM can help in troubleshooting deployment scenarios such as clientless-mode and other programmability options.

The following figure shows a simplified protocol flow for a typical browser-based client-side interaction with BIG-IP APM.

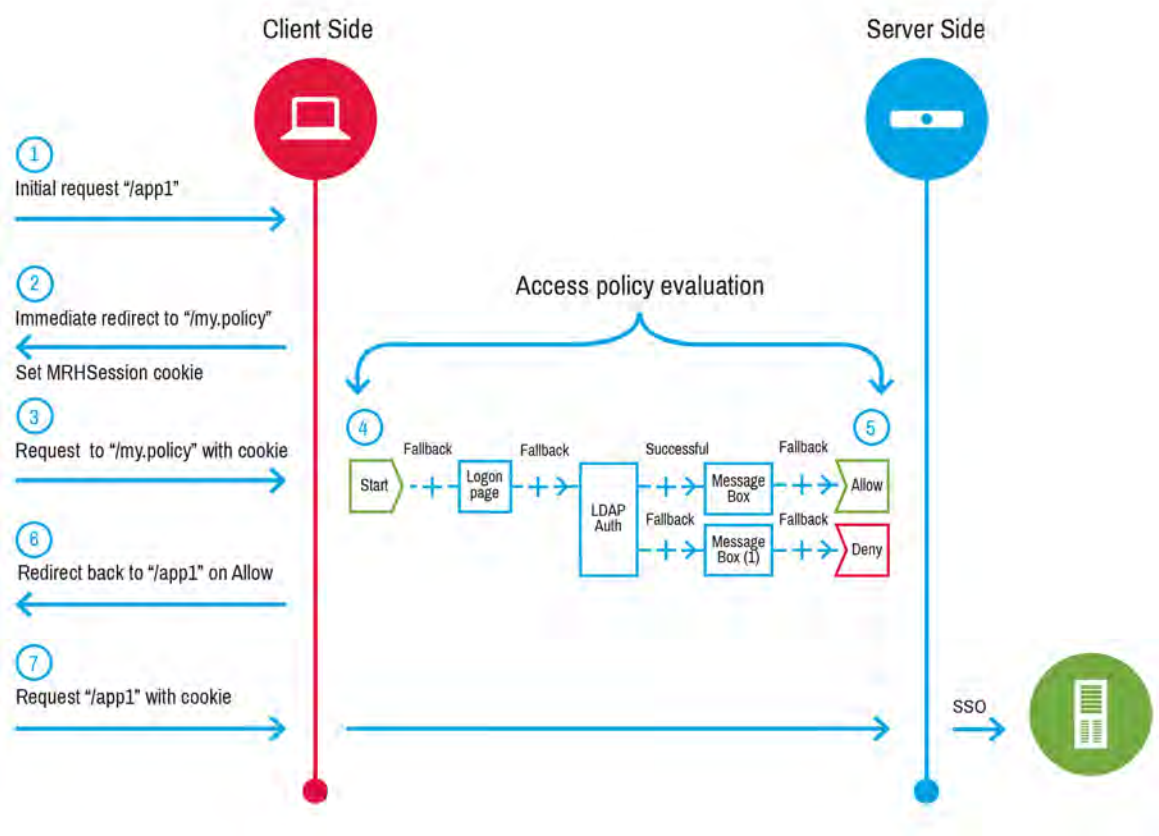

Figure 1.1 Client interaction with BIG-IP APM

1. The client makes an initial request to a BIG-IP APM virtual server. The request may have no specific URI, in which case the URI is **"/"** or a unique URI pattern, as in shown in the figure.
2. BIG-IP APM creates an access session.
   The client is redirected to a **"/my.policy"** URI.
   A session cookie (pointer) for that access session is set in the redirect response.
3. Client browser returns request to **/my.policy** and BIG-IP APM session cookie, **MRHSession**.
4. Access session enters starts and BIG-IP APM begins access policy evaluation.
   Policy agents such as the **Logon Page** or **Message Box** may send responses to client.
5. If access policy evaluation ends at **Deny**, the access session is marked "denied" and BIG-IP APM terminates the session and responds with a customizable error page.
   If access policy evaluation ends at **Allow**, the access session is marked as "allowed."
6. If the session is marked as "allowed," BIG-IP APM redirects back to the original request URI.
7. Client browser returns to the URI with the session cookie.
   Access policy evaluation is skipped, and single sign-on (SSO)—if applied to the access policy—is enabled.
   All following requests with this session cookie to the BIG-IP APM virtual IP will skip access policy evaluation. SSO will remain enabled to maintain the server-side authenticated state.
   Session may expire, depending on configuration of session options.

BIG-IP APM with other BIG-IP modules

With the introduction of the F5 Good, Better, Best licensing and provisioning model, the BIG-IP platform provides the ability to license and provision multiple software modules. Various module combinations can be utilized to meet the specific needs for the network environment. The ability to provision multiple software modules is the foundation for implementing *F5 Reference Architecture* solutions.

BIG-IP APM is capable of working with the following BIG-IP modules:

- BIG-IP Global Traffic Manager™ (GTM)
- BIG-IP Application Security Manager™ (ASM)
- BIG-IP Advanced Firewal Manager™ (AFM)
- BIG-IP Application Acceleration Manager® (AAM)

When combined, these modules work together to enhanced redundancy, security, and performance. For more information about Good, Better, Best licensing, see AskF5 article: ***SOL14826:*** *Good, Better, Best license options and provisioning*.

Note Module combinations are limited by the amount of platform system memory. For more information about module compatibility, refer to the BIG-IP system software version's release note.

BIG-IP GTM

BIG-IP GTM and BIG-IP APM can be used together to provide high availability and secure remote access to corporate resources from anywhere in the world. BIG-IP GTM can be configured to intelligently direct traffic to the available branch office closest to the user. The BIG-IP APM uses one of several options to authenticate the user and then creates a secure session between the user and the remote office.

There are two topologies that can be used to deploy a BIG-IP GTM and BIG-IP APM solution:

- High availability configuration
- Topology-based configuration

For more information, see ***Deploying BIG-IP GTM with APM for Global Remote Access***.

BIG-IP GTM, BIG-IP LTM, and BIG-IP APM can be used together to provide a single namespace (for example, https://desktop.example.com) to clients accessing VMware Horizon with View virtual desktops.

BIG-IP GTM and BIG-IP LTM work together to ensure that requests are sent to a user's preferred data center, regardless of the user's current location. Additionally, BIG-IP APM validates the login information against the existing authentication and authorization mechanisms such as Active Directory, RADIUS, HTTP, or LDAP.

BIG-IP ASM

BIG-IP ASM and BIG-IP APM can be used together to track sessions using authentication provided by a BIG-IP APM access policy and using BIG-IP ASM session tracking. These modules when used with database security products, such as IBM InfoSphere Guardium, to increase security visibility, receive alerts about suspicious activity, and prevent attacks.

For more information, see *Tracking Application Security Sessions with APM* and *Overview: Integrating ASM and APM with database security products* in ***BIG-IP Application Security Manager: Implementations***.

BIG-IP AFM

BIG-IP AFM can be used in application delivery controller mode, which allows traffic to virtual servers and self IPs on the system. Any traffic you want to block must be explicitly specified. BIG-IP AFM is a network firewall and applies only to the virtual server and self IPs on the system.

BIG-IP AFM can also be deployed in Firewall mode, which applies a default deny policy to all self IPs and virtual servers. In this mode, to allow access to BIG-IP APM, firewall rules must be created at the virtual server level.

BIG-IP AFM rules do not apply to VPN tunnel traffic from VPN clients to internal networks. F5 recommend using the built-in ACL in BIG-IP APM.

For more information, see *Configuring Network Access Resources* in ***BIG-IP Access Policy Manager: Network Access***.

BIG-IP AAM

BIG-IP AAM can be used in together with BIG-IP APM Portal Access to provide:

- **Improved performance for an HTTP or HTTPS stream** by offloading the compression overhead from origin web servers.
- **Caching of patched (rewritten) Portal Access objects** and their delivery directly to clients, improving web page loading time due to repeated file patching.

As a general recommendation, an HTTP compression and a web acceleration profile should be applied to a BIG-IP APM virtual server. This will make sure that all content delivered to the client is compressed. The addition of a Web Acceleration profile ensures that files are not repeatedly patched over and over.

For more information see ***BIG-IP AAM 11.6.0 Documentation***.

Licenses

Introduction

License types

License limits

BIG-IP APM Lite

Introduction

BIG-IP APM session licensing is handled within the BIG-IP licensing infrastructure.

For more information, see AskF5 article: ***SOL7752: Overview of licensing the BIG-IP system***.

BIG-IP APM license types

BIG-IP APM uses two different types of licenses:

- **Access session licenses**, which are consumed when a user starts any new session.
- **User connectivity licenses (CCUs)**, which are consumed when a user is assigned one or more BIG-IP APM resources with tunnel-type access.

Access session licenses

When a user connects to BIG-IP APM, a new session starts and an access session license is used. Once a license is used, it cannot be used again until the user session terminates.

After the access session begins, it is subjected to access policy evaluation, with one of two outcomes possible:

- **Access policy evaluation succeeds,** and the access session license remains unavailable for other sessions until the current session is terminated or the user logs out.
- **Access policy evaluation fails**, the session is terminated and the access session license is released and made available for a new session.

Note Applications running the LTM-APM profile type for web application access (such as Microsoft SharePoint) consume one access session license.

Important Exceeding the maximum licensed session count will lead to loss of service. The number of access session licenses available is determined by the platform on which BIG-IP APM is running.

An additional add-on SKU is available to maximize the number of access session licenses available for the platform. This license is specific to the capabilities of the given hardware platform. Maximum license counts for different platforms are listed in the ***BIG-IP APM Data Sheet***.

For more information, see ***BIG-IP APM Datasheet***.

The following figure shows license consumption process in BIG-IP APM user session management.

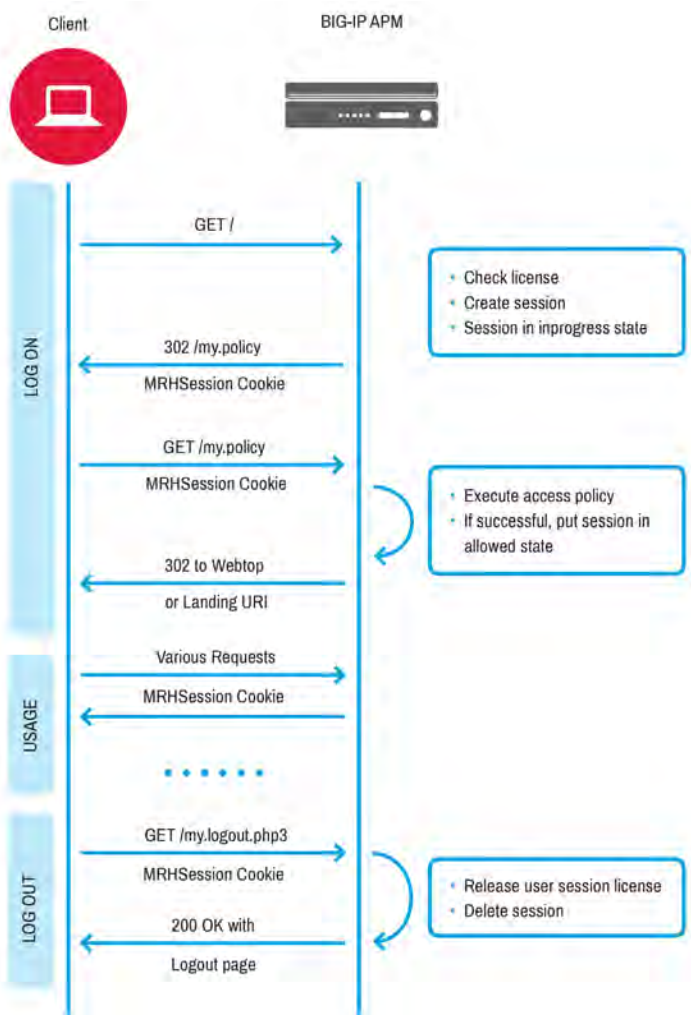

Figure 2.1: BIG-IP APM license consumption overview

CCU licenses

A CCU license is used when a user is assigned one or more BIG-IP APM resources which have tunnel-type access processed by passing through BIG-IP APM. No matter how many resources are assigned to the user, each session consumes only one CCU license.

For example, if a user has access to a BIG-IP APM full webtop with network access and an application tunnel, then one CCU license is consumed per access session. If the user has access to a BIG-IP APM full webtop which does not contain any CCU resources, then no CCU licenses will be consumed.

Note An access session license is always consumed for any session.

The following table shows which access resource types require a CCU license in addition to an access session license:

Table 2.1 License requirements by resource type

Access Category	Specific Access	CCU
Network access (L3)	BIG-IP Edge Client, browser, API	Yes
Application access	Citrix clients (Proxy mode, Web interface replacement mode [webtop])	No
	VMWare View	No
	MS Remote Desktop Protocol client (ActiveX or Java)	No
	Application tunnel	Yes
Portal access	Web-based application on webtop (such as OWA,Microsoft SharePoint, XenApp)	Yes
	Citrix portal mode (Web interface)	Yes
Other	Exchange (OWA, ActiveSync, EWS, Outlook Anywhere, and others)	No
	Web authentication	No
	Oracle Access Manager	No
	SAML resource on webtop	No

For more information about license types, see AskF5 article: ***SOL13267: BIG-IP APM connectivity license***.

License limits

To view the number of access licenses using tmsh at the command line

- Type the following command:

```
tmsh show /sys license detail | grep apm_access_sessions
```

  The command output appears similar to the following example:

```
apm_access_sessions [2500]
```

  In this example, a total of 2500 total access licenses are available for use.

To view the number CCU licenses using tmsh at the command line

- Type the following command:

```
tmsh show /sys license detail | grep apm_sessions
```

  The command output appears similar to the following example:

```
apm_sessions [250]
```

  In this example, a total of 250 total CCU licenses are available for use.

For more information, see AskF5 article: ***SOL15032: Determining license limits of the BIG-IP APM system***.

If a user attempts a new connection when the session limit has been reached, two actions occur:

- The following error message is logged to **/var/log/apm**:

```
 warning tmm1[10186]: 01490508:4: 00000000: Global concurrent access session limit
reached.
```

- The user is redirected to the login page, where the following error message is displayed:

```
 The maximum number of concurrent user sessions has been reached. No new user
sessions can start at this time.
```

BIG-IP APM Lite

All BIG-IP systems include a free perpetual license for the BIG-IP APM Lite module.

This module includes the same features as a fully licensed BIG-IP APM module, with the following limitations:

- Licenses are limited to 10 each for access sessions and CCU sessions.
- Hardware compression is disabled.
- Software compression is limited to 50Mbps.
- Authentication, Authorization, and Accounting (AAA) Oracle Access Manager (OAM) integration is not provided.

Use cases

Introduction

Authentication and single sign-on

Network access

Per application VPN

Application tunnel

Web access management

Portal access

Citrix integration

VMware View support

Remote Desktop Protocol support

Exchange proxy

Webtop

Access control list

Introduction

BIG-IP APM manages secure remote access for network applications and clients. It can be configured and deployed to provide a variety of access management functions, including:

- Authentication and single sign-on.
- VPN (network access, per-app VPN).
- Tunnels (application access).
- Secure web access (LTM-APM).
- Reverse HTTP proxy (portal).
- Citrix, VMware View, and RDP access.
- Microsoft Exchange services access.

The following sections describe several common BIG-IP APM use case options, including information regarding features, required components, and implementation.

Authentication and single sign-on

BIG-IP APM serves as an authentication gateway or proxy. As an authentication proxy, BIG-IP APM provides separate client-side and server-side authentication. The client-side authentication occurs between the client and BIG-IP APM. The server-side occurs between BIG-IP APM and backend servers.

Loose coupling between the client-side and server-side layers allows for a rich set of identity transformation services. Combined with a Visual Policy Editor and an expansive set of access iRules functionality, BIG-IP APM provides flexible and dynamic identity and access, based on a variety of contexts and conditions.

For example, a client accessing Microsoft SharePoint through BIG-IP APM in a corporate environment may silently authenticate to BIG-IP APM with NT LAN Manager (NTLM) or Kerberos credentials. On leaving that environment, or on using a different non-sanctioned device, the client may be required to go through another potentially stronger authentication, such as a smartcard or other client certificate, RSA SecurID, or one-time passcode. Additional device vetting such as file, folder, and registry checks and antivirus and firewall software validation can be required.

A BIG-IP APM authentication and single sign-on (SSO), access and identity security posture can automatically change depending on environmental factors such as who or where the user is, what resource the user is accessing or when, or with what method the user is attempting to gain access.

Features

There are several reasons to use BIG-IP APM authentication and single sign-on (SSO), including pre-authentication, dynamic access and identity control, and authentication gateway and identity transformation.

Pre-authentication

BIG-IP APM pre-authentication adds an additional layer of application security by dynamically authenticating and validating users before allowing access to the back-end resources. Pre-authentication can be used for standard web access or other BIG-IP APM access use cases, including network access, portals, application tunnels, and virtual desktop infrastructure resources.

As a pre-authentication service, BIG-IP APM can enforce stronger authentication processes than the back-end services can natively support. For example, BIG-IP APM can be deployed to require client certificate or other two-factor token methods in front of applications that only support Kerberos. Or an RSA SecurID passcode can be added as a second factor of authentication to applications that only support username/password authentication.

The following figure shows the interaction between a client, BIG-IP APM, and the SSO functions. It shows how user credentials are exchanged with internally hosted applications using SSO.

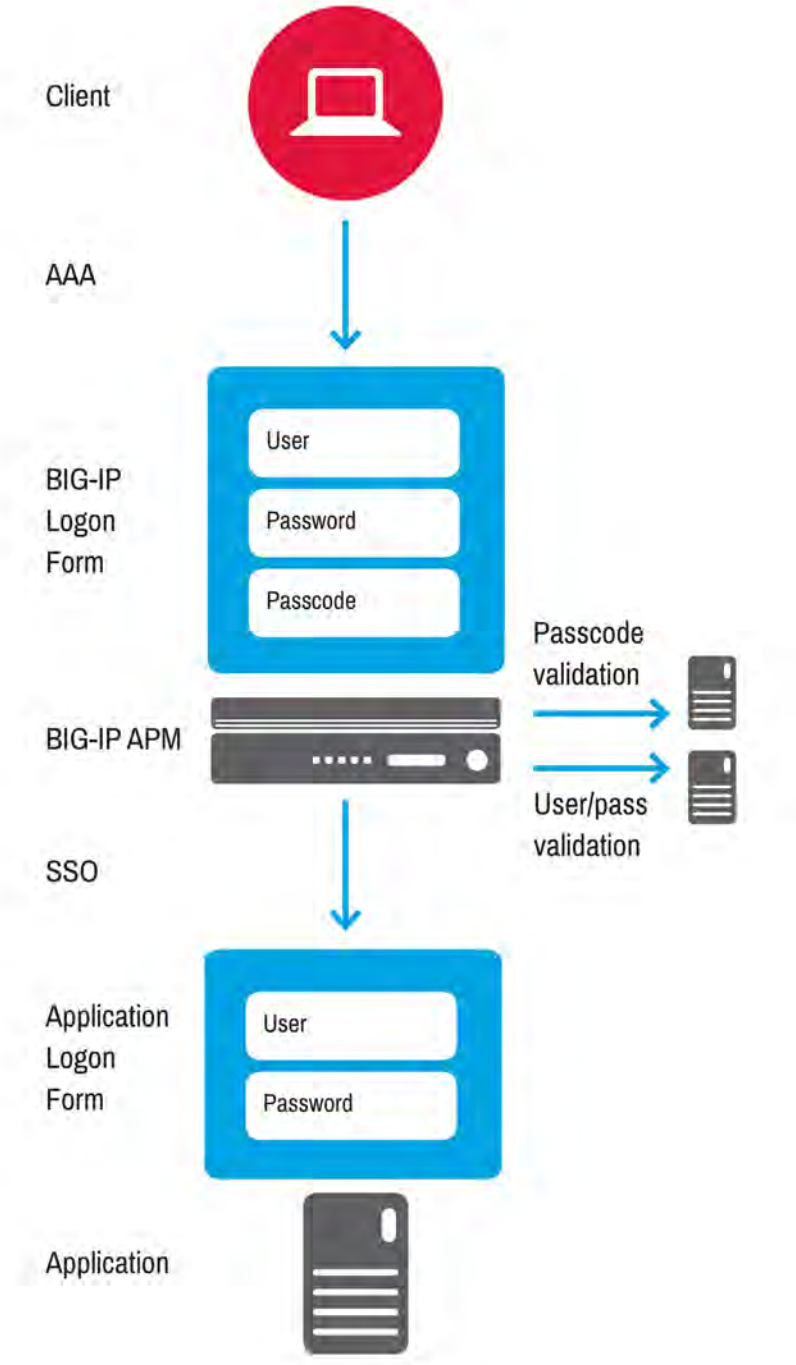

Figure 3.1: Pre-authentication and SSO

Dynamic access and identity control

BIG-IP APM can change both the protocol by which a client asserts identity information, and the ways in which that identity information is validated, based on environmental factors.

In the SharePoint example described in the introduction to this section, internal corporate users can present Kerberos or NTLM credentials to BIG-IP APM for access. On leaving the corporate environment, the access policy can be configured to enforce different and/or additional authentication methods, such as client certificate or one-time passcode. It can also insert additional endpoint posture checking like antivirus and system service checks.

The following figure shows how BIG-IP APM validates client identity based on environmental factors and provides a stronger authentication layer.

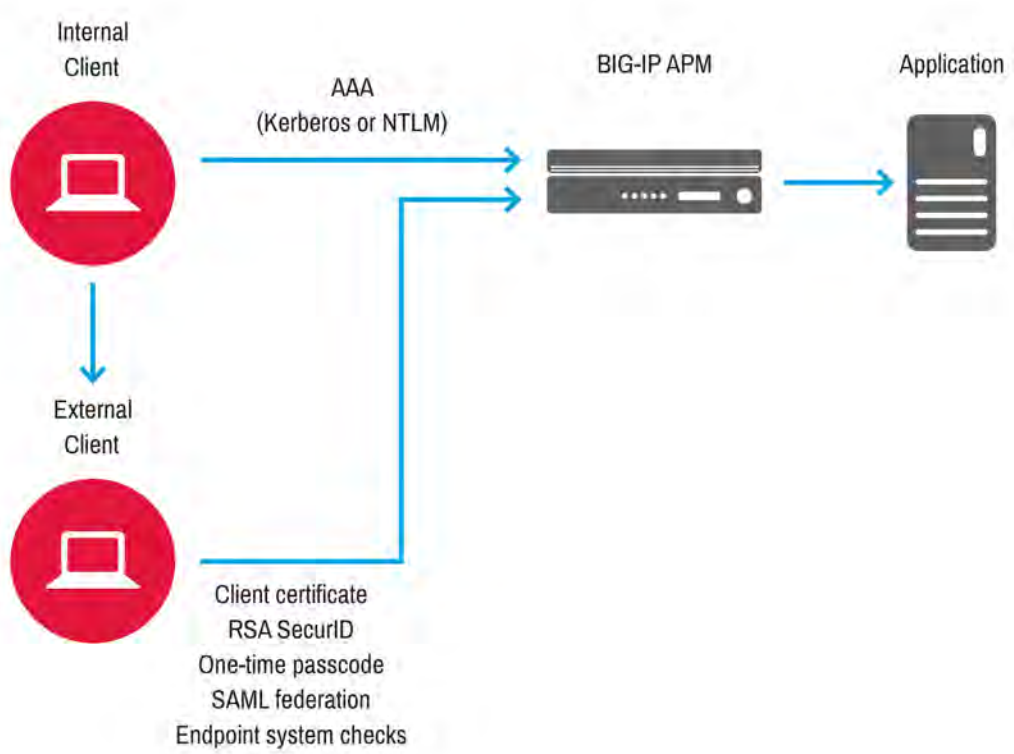

Figure 3.2: BIG-IP APM client identification

Authentication gateway and identity transformation

Data centers often face the challenge of offering multiple applications with different authentication requirements. BIG-IP APM can be deployed to consolidate and enforce all client-side authentication into a single process. It can also perform identity transformation on the server side to authenticate to back-end services in the methods that are best supported. This can reduce operational costs since applications remain in the most supported and documented configurations. Common examples of identity transformation are client-side PKI certificate to server-side Kerberos and client-side HTTP form to server-side HTTP Basic.

The following figure shows BIG-IP APM acting as an authentication gateway. Information received during pre-authentication is transformed to authenticate to multiple enterprise applications with different requirements.

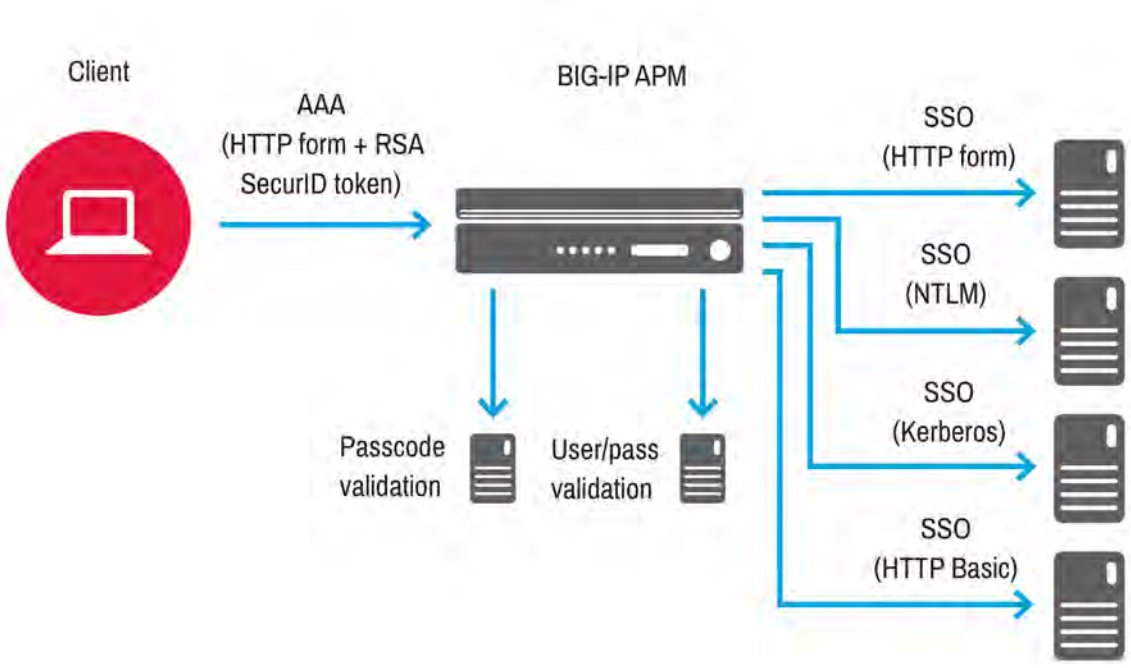

Figure 3.3: BIG-IP APM as an authentication gateway

Components

BIG-IP APM authentication and single sign-on components can be grouped into separate client-side and server-side functionalities.

Client-side authentication involves the client (generally a user with a browser) accessing a BIG-APM virtual server and presenting identity. This is called authentication, authorization, and accounting (AAA).

Server-side authentication involves BIG-IP APM providing authentication to a back-end resource. This is called single sign-on (SSO).

Client-side authentication

BIG-IP APM supports industry standard authentication methods, including:

- NTLM
- Kerberos
- SAML
- Client certificate
- RSA SecurID
- One-time passcode
- HTTP Basic
- HTTP Form

Once access credentials are submitted, BIG-IP APM validates these with industry standard mechanisms, including:

- Active Directory authentication and query
- LDAP and LDAPS authentication and query
- RADIUS
- TACACS
- OCSP and CRLDP (for client certificates)
- Local User Database authentication

BIG-IP APM can further vet client access by inspecting the client device itself, using methods including (but not limited to):

- File system checks

- System service checks
- Registry checks
- Browser plugin checks
- Antivirus software checks
- Firewall software checks
- Hard-disk encryption software checks
- Patch management software checks
- Peer-to-peer software checks
- Hardware certificate checks
- OS and client device ID checks

Authentication, validation, and vetting mechanisms are defined within the access policy. For more information on creating BIG-IP APM client-side authentication functionality, see ***BIG-IP Access Policy Manager: Authentication and Single Sign-On***.

Server-side authentication

Client side and server side are loosely coupled in the authentication proxy. Because of this, client-side identity values of one type can be transformed into server-side identity values of another type. SSO is configured as within an SSO profile, which is applied to an access profile. It is triggered at the end of successful access policy evaluation and on subsequent client-side requests.

BIG-IP APM supports industry standard authentication methods, including:

- NTLM
- Kerberos
- HTTP Basic
- HTTP Form
- SAML

Note Client-side authentication methods outnumber server-side methods. This is because BIG-IP APM does not transmit client certificate, RSA SecurID, or one-time passcodes to the server on the client's behalf.

For more information on BIG-IP APM server-side authentication, see ***BIG-IP Access Policy Manager: Authentication and Single Sign-On***

Compatibility

A successful BIG-IP APM client-side authentication will produce a set of session variable values as output that server-side authentication can consume and use as input. However, some server-side methods have requirements that the client side cannot fulfill.

For example, server-side NTLM is a challenge-response mechanism. It requires knowledge of the user's password. A client-side Kerberos authentication would not provide access to the user's password, so these two methods are incompatible.

The following table shows the compatibility between various client-side and server-side authentication methods.

Table 3.1: **Client-side and server-side authentication method support matrix**

Client-side Authentication Methods	Server-side Authentication Methods: HTTP form	HTTP Basic	NTLM	Kerberos
HTTP form	Yes	Yes	Yes	Yes
HTTP Basic	Yes	Yes	Yes	Yes
NTLM	No	No	No	Yes
Kerberos	No	No	No	Yes
Certificate	No	No	No	Yes
SAML[1]	Yes[1]	Yes[1]	Yes[1]	Yes
RSA SecurID	No[2]	No[2]	No[2]	Yes
One-time passcode	No[2]	No[2]	No[2]	Yes

1. BIG-IP APM can function as a SAML identity provider (IdP) and a service provider (SP). As an SP, the client will generally authenticate at the IdP. Therefore, the SP will not have access to user's credentials. However, it is possible for the IdP to transmit those validated credentials, encrypted, in the standard SAML assertion or in a separate artifact communication. In this way a BIG-IP APM SAML SP could perform server-side authentication functions requiring a password.

2. RSA SecurID and one-time passcode are rarely used alone. They are usually combined with username and password authentication to add an additional authenication factor. If these methods are combined with a username and password prompt, then they collectively support server-side authentication methods that require a password. However, if these methods are used in a capacity to replace a user password, they generally cannot support server-side authentication methods requiring a password.

Network access

BIG-IP APM network access supports full OSI layer 2 remote access VPN connectivity to internal network resources. Network access resources assigned to an access policy provide a wide array of security and optimization capabilities for both desktop and mobile clients.

With BIG-IP APM network access, once connected, the internal network is available to the client. Other controls and features are available to support variations.

Features

The network access features listed in the following table can be configured for each network access resource created.

Table 3.2 Network access features

Feature	Description
Compression	Enabled in network access resource. Parameters of GZIP compression are configured in the connectivity profile. Compression typically provides little benefit, as most network traffic is pre-compressed.
Forward error correction	Licensable module included with BIG-IP Application Acceleration Manager®. Forward error correciton (FEC) provides reliability for Datagram Transport Layer Security (DTLS) tunnels at the cost of higher bandwidth usage. It saves bandwidth by enabling compression on DTLS traffic and reducing or eliminating TCP retransmissions.
SNAT selection (from policy or NA resource)	Network access uses flexible mechanisms to choose source-NAT addresses based on access policy parameters. Generally SNAT is disabled to support VoIP and improve reliability of SMB similar protocols.
Routing domains	Provide the capability to segment network traffic and define separate routing paths for different network objects and applications.
Bandwidth controller policy	Allows for static or dynamic bandwidth control per user.

Feature	Description
Access control lists (ACLs)	For information on ACLs, see the access control lists section in this chapter.
Application launch	Allows for a client application to be launched immediately after a network access connection is established. For example, launching a web browser to a company intranet portal. For more information, see ***Launching application on network access connection.***
Reconnect to domain	Synchronizes Active Directory policies and executes domain logon scripts in domain-joined Windows client PCs. It also enables a second option to execute logoff scripts, if desired.
Drive mapping	Allows for network drives to be mapped after establishing the network access connection.
On-demand VPN	Allows the mobile BIG-IP Edge Client to launch automatically, given specified URLs. This is typically used with transparent client certificate authentication to allow seamless access.
SSO	Can be transparently applied to VPN user traffic. When using SSO, ACLs will not be processed for the resources defined for the internal virtual server(s). For more information, see AskF5 article: ***SOL11312: Creating network access with single-sign on capabilities***
Proxy support	If clients use internal proxy for web access, BIG-IP APM allows for flexible options to apply a static proxy server or PAC configurations to VPN clients that connect. If clients use proxy to access BIG-IP APM to create a tunnel, the VPN client supports the condition that the VPN tunnel is created through a proxy server

Important In application launch, if more than one operating system is used, pay close attention when specifying application paths and launch parameters.

Split tunneling and DNS

BIG-IP APM supports split tunnels as well as full tunnels. The network access client, including BIG-IP Edge Client, changes the client's routing table based on the network access resource configuration. Multiple routes, or a default route, can be used to direct the client's traffic through the tunnel. For more information see *Network Security* in the *Security* chapter of this guide.

The Windows network access client, including BIG-IP Edge Client, has a flexible proxy DNS service. The DNS service can forward client DNS requests to BIG-IP APM for processing. BIG-IP APM can then answer the requests directly or forward them to the local DNS server.

Components

The following BIG-IP APM components must be created before network access can be assigned to users:

- A connectivity profile.
- A network access lease pool to assign to connecting clients.
- A network access resource to configure network access properties.
- A full webtop or network access webtop to present the network access resource to the client.
- An access policy that assigns the webtop and network access resource.

In addition, a VPN web client must be (automatically) downloaded into the user's browser, or the stand-alone BIG-IP Edge Client must be pushed out to the user by the BIG-IP administrator.

For more information about which features are available on which operating systems, see ***APM Client Compatibility Matrix.***

Implementation

There are two types of transport uses for BIG-IP APM network access: Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS). TLS uses TCP as the transport. DTLS uses UDP. DTLS has lower overhead than TCP and may be better suited for VoIP and VDI solutions. Both TLS and DTLS network access work in either an IPv4-only environment or a mixed IPv4 and IPv6 environment.

BIG-IP APM also supports network access optimized applications. Optimized applications are configured in the Network Access feature and allow layer 4 tunnels to internal networks using the same iSession transport method as application tunnels. Application tunnels and network access optimized applications support various compression codecs as follows:

- Client to server codecs are configured in the connectivity profile.
- Server to client codecs are configured in the Network Access feature, as are the internal TCP network/subnet endpoints.

Three compression methods are supported: deflate, LZO, and bzip2. Adaptive compression automatically selects the compression type based on network and traffic characteristics.

The following figure shows an overview of client-server interaction during layer 2 VPN tunnel estabilshment.

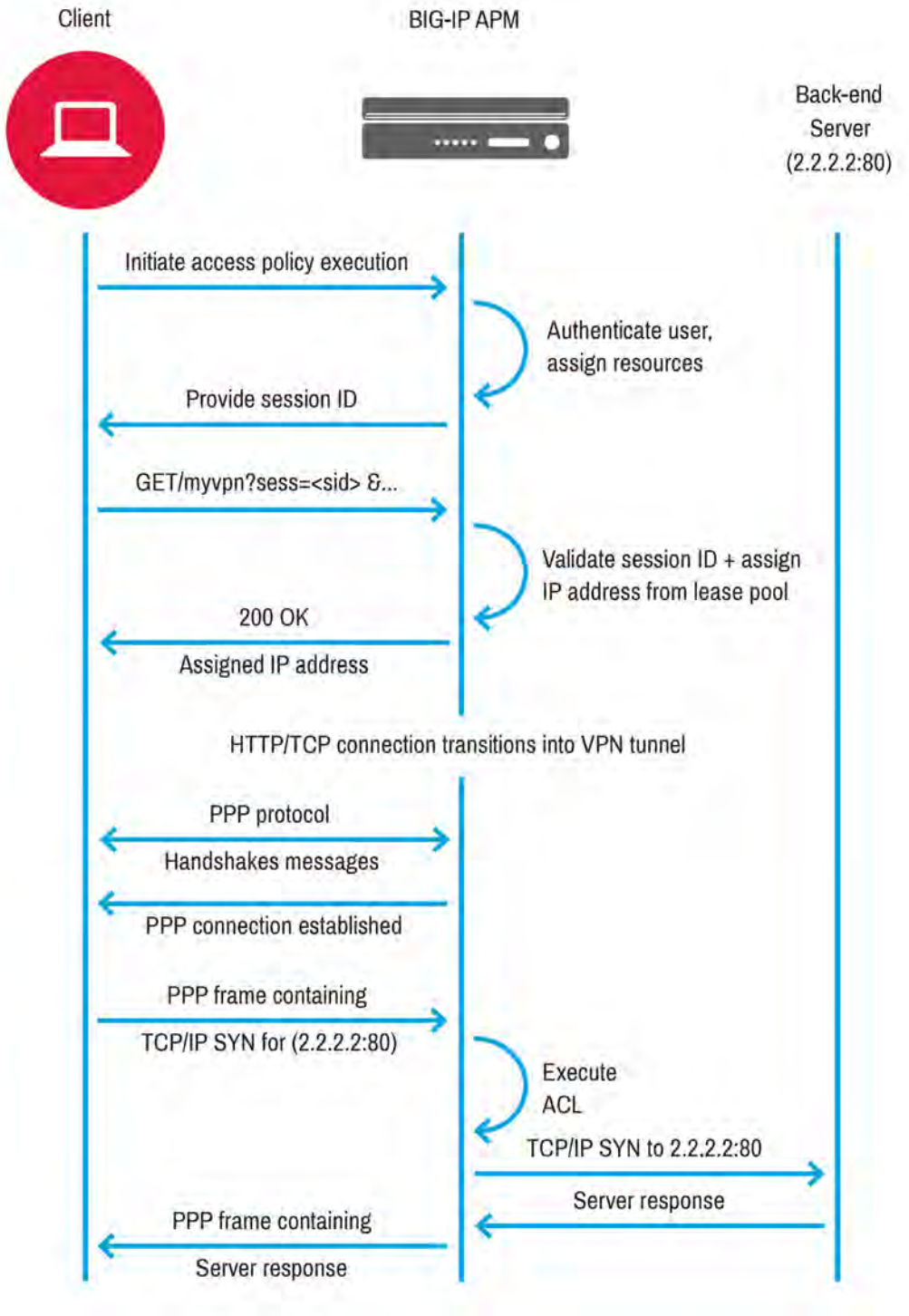

Figure 3.4: Establishing a VPN tunnel

1. A user browses to a virtual server URL from a BIG-IP APM client and initiates an access policy evaluation sequence.
2. After successful access policy evaluation, a valid session is created and the network access resources are assigned.
3. The client requests a VPN configuration using an HTTP GET method.
4. The client establishes a layer 2 tunnel with BIG-IP APM.
5. All of the traffic destined for BIG-IP APM is encapsulated in layer 2 frames.

Per-application VPN

A per-application (per-app) VPN makes sure that specific mobile applications and their data remain secure and protected, and only data relevant from the application is sent to the internal network. With the per-app VPN capabilities of the BIG-IP APM, combined with a mobile device management (MDM) solution, enterprise organizations can be sure only authenticated and authorized mobile users are able to access and send data to the organization from approved mobile applications or mobile containers.

Features

Per-app VPN deployed using an existing MDM solution. Depending on the authentication used, it can offer a seamless or relatively simple way to access internal resources. Per-user bandwidth policies and ACLs can be applied to make sure that users comply with network use policies. Detailed user activity auditing is also possible with ACL logging or solutions based on iRules.

The access policy can be configured to act on various mobile properties, such as device ID, compliance status, and more, to provide granular control on per-app VPN tunnel connections.

Components

The following BIG-IP APM components must be created before per-app VPN can be assigned to users:

- A connectivity profile.
- An application tunnel profile (Java and per-App VPN).
- An MDM-enrolled mobile devices.
- An MDM-deployed application.
- The BIG-IP Edge Client 2.0.1+ for layer 3 and per-app VPN.

Note Per-app VPN functionality is supported on iOS and Android Edge Clients version 2.0.7 and higher.

Implementation

Per-app VPN tunnels use an F5 proprietary version of the SOCKS protocol.

The following figure shows a per-app VPN tunnel packet flow.

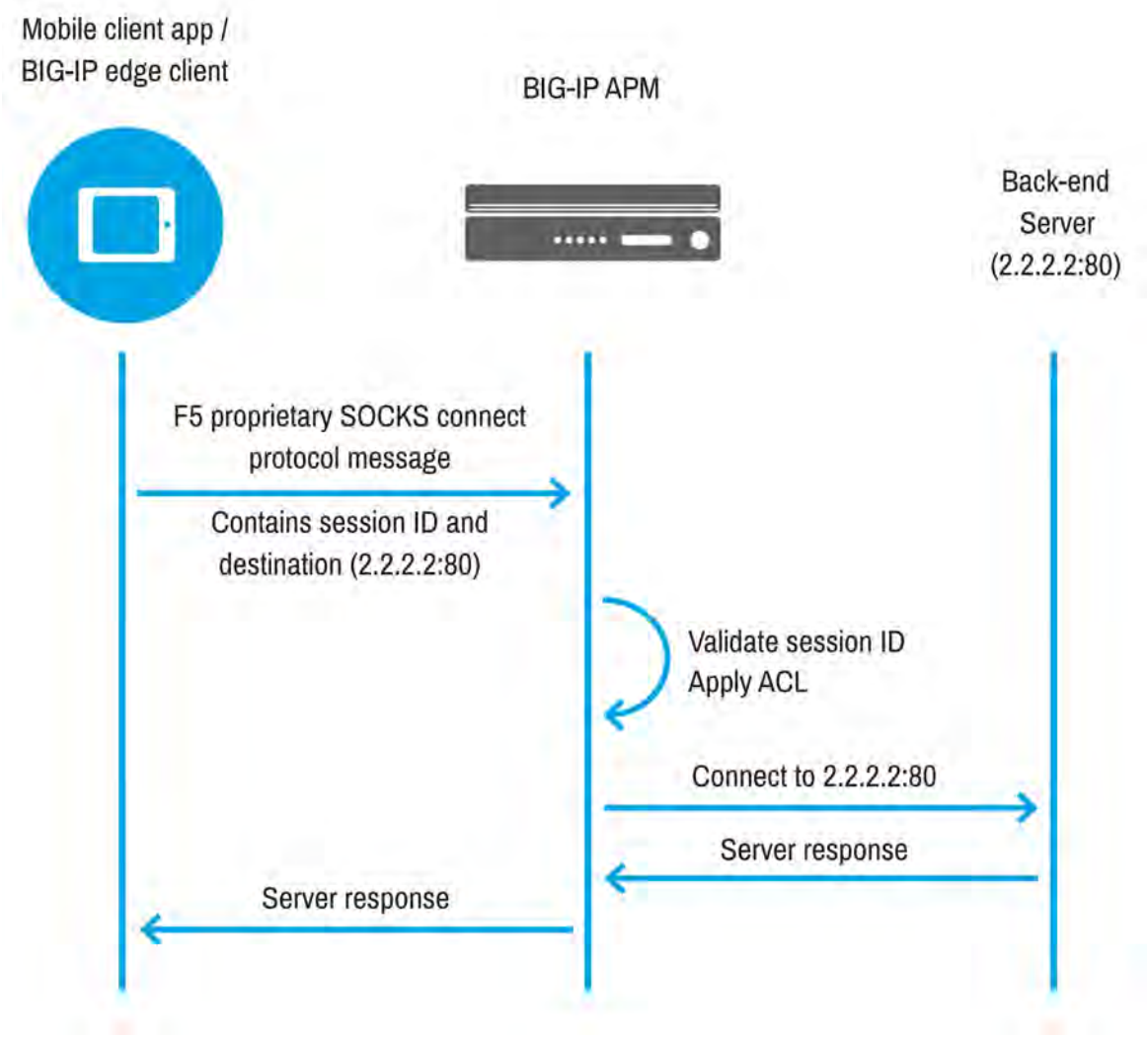

Figure 3.5: Per-app VPN tunnel packet flow

For more information on per-app VPN, see ***AirWatch/F5 Solution for Enterprise Mobility***.

Application tunnel

An application tunnel (app tunnel) provides secure, application-level TCP/IP connections from the client to the internal network.

Features

App tunnels can be used to provide access for users with limited privileges who need to access internal applications. App tunnels do not require administrative privileges to install client modules.

App tunnels have lower overhead in connection establishment, lower client module complexities, and faster application connections when compared to network access. Unlike network access, app tunnels allow simultaneous creation of multiple connections from a client, even to different BIG-IP APM endpoints.

App tunnels use iSession, an F5 proprietary protocol for transport. App tunnels can be launched using native Windows binary components or with a browser-based Java applet on Windows, Mac and Linux platforms. Per-user session-level bandwidth policy and access control lists (ACLs) can be applied to app tunnels.

App tunnel optimization

Optimization is available for app tunnels. Available compression codecs settings for client-to-server connections can be configured on an app tunnel resource. The server compares the available compression types with the available compression types on the client, then chooses the most effective mutual compression setting. The compression settings for the client can be configured in the connectivity profile and the compression settings for the server can be configured in the app tunnel resource.

Note It is recommended to configure application parameters in an application tunnel resource item with %host% and %port% parameters. Sometimes due to local loopback port conflicts with other applications, app tunnels may be created on different local ports and %port% is updated with the correct port in that case. The %host% parameter translates to http://%host%/application and is needed in case the user doesn't have enough privileges to update the static host file.

Components

An application tunnel is comprised of the following components:

- A connectivity profile.
- A full webtop.
- An application tunnel resource.
- An access policy that assigns a webtop and an application tunnel resource.

The following figure shows client-server communication sequence when the application tunnel is being established. After successful access policy execution, client sends HTTP **GET/isession?sessionid=<sid>** requests to setup iSession with BIG-IP APM.

Client then establishes iSession connection with BIG-IP APM, and all future communication thereafter is encapsulated in F5 proprietary frames.

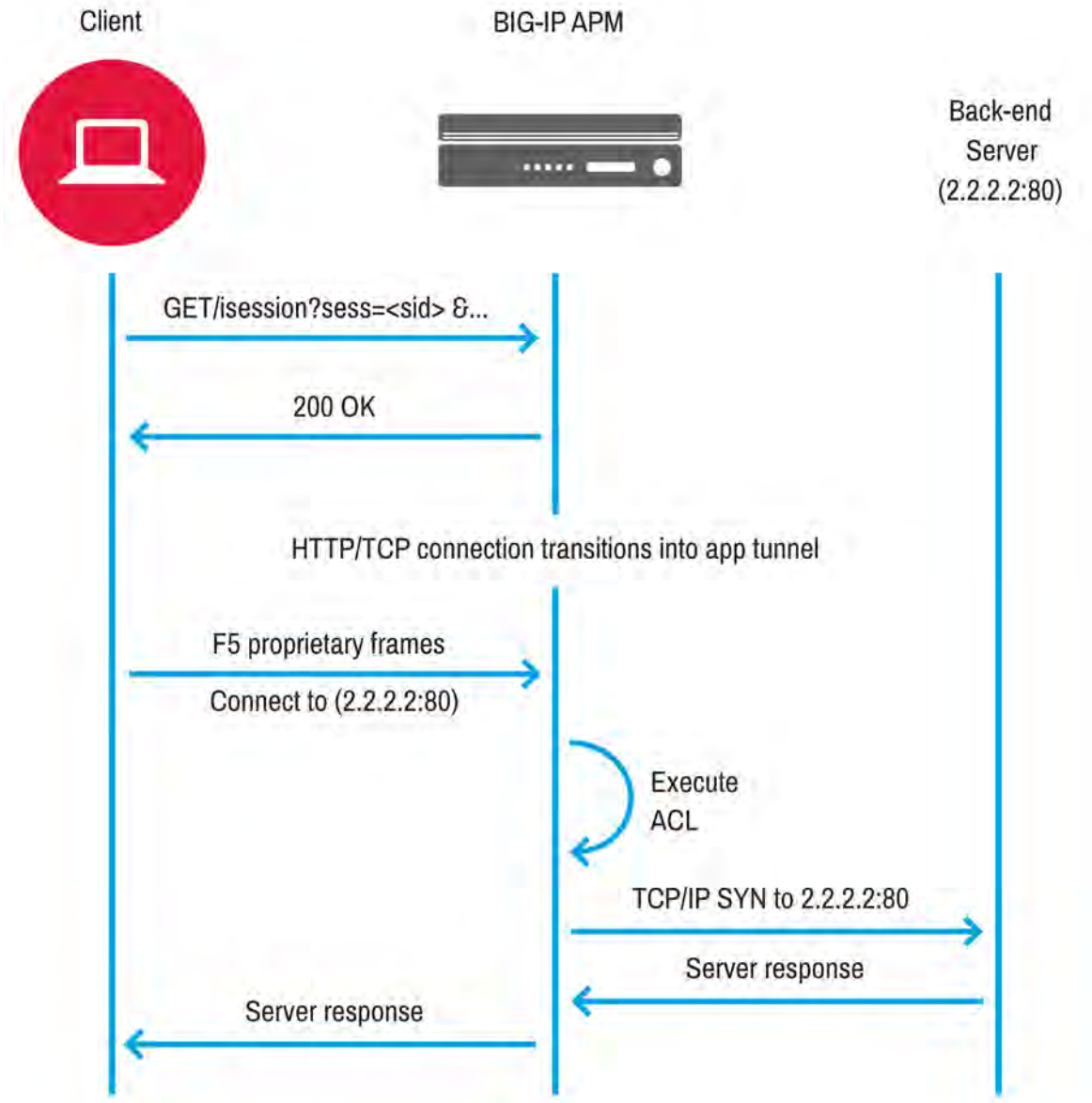

Figure 3.6: Application tunnel packet flow

For more information regarding application tunnels, see ***Introduction to iSessions*** on DevCentral and the ***Configuring App Tunnel Access*** chapter of the ***BIG-IP Access Policy Manager: Application Access Guide***.

Web access management

The BIG-IP LTM module manages and optimizes traffic for network applications and clients. It can automatically load balance application traffic amongst multiple internal servers. The back-end server's SSL encryption can be offloaded to the BIG-IP. BIG-IP APM can be integrated with BIG-IP LTM to provide authenticated access to web applications through a web browser without the use of tunnels or specific resources

Features

Web applications that do not provide native user login and account validation can be protected using the web access management feature. In some cases this can reduce the expense of application development and deployment.

- BIG-IP APM provides a wide range of authentication mechanisms allowing flexible deployment and secure access to the backend resources.
- BIG-IP APM can provide custom user auditing using built-in or iRules-based functionality.

Note Web access management is also called LTM+APM and LTM-APM in various F5 documentation.

Components

The following components must be created in order to use web access management:

- An access policy configured with an authentication agent.
- A pool of resources.
- An LTM virtual server.

Implementation

The following figure shows communication between the client and server when the client attempts to access protected web application resources.

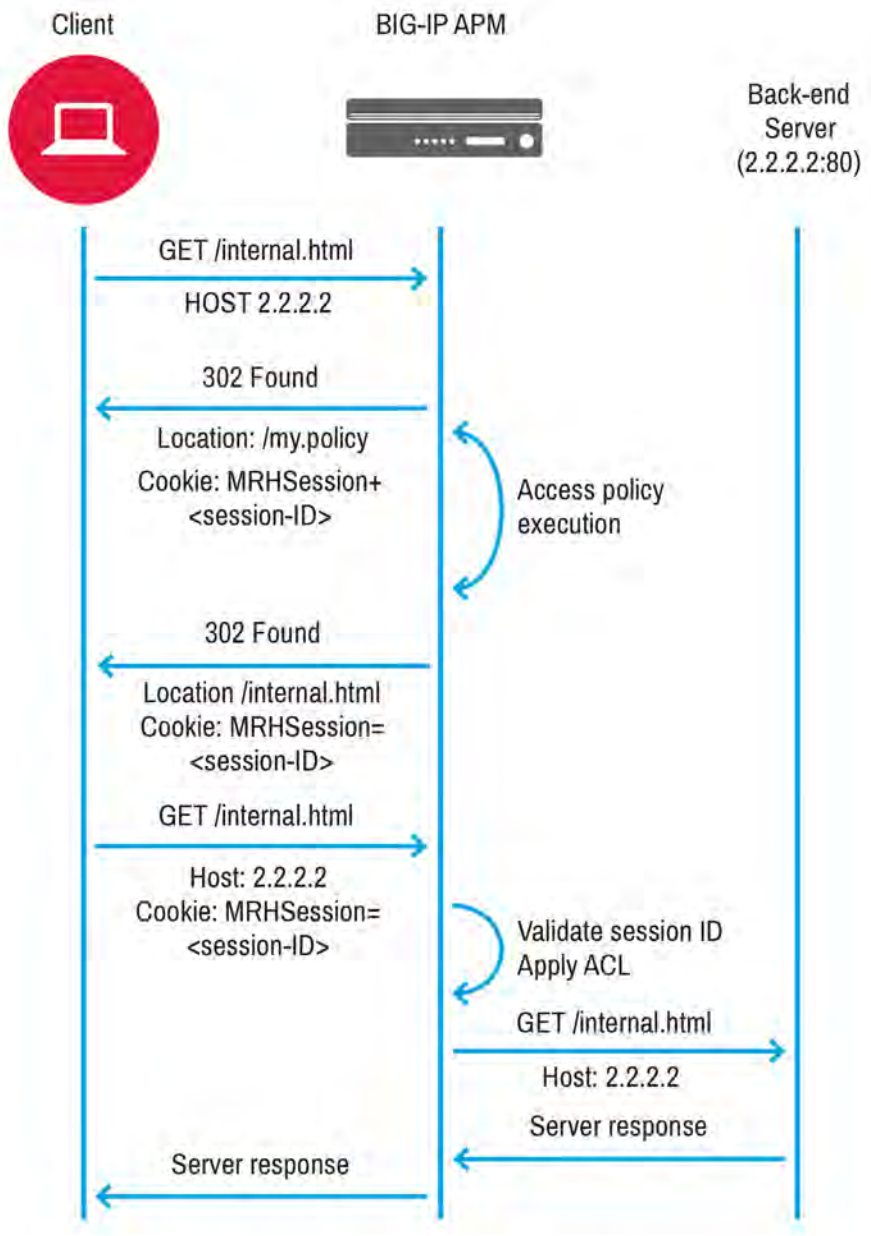

Figure 3.7: Web access management packet flow

Portal access

Portal access is the HTTP reverse proxy feature for BIG-APM. It allows for any number of internal hosts to be accessed remotely.

A rewrite process is implemented to retrieve content on the user's behalf. Web content including HTML, Java, JavaScript, CSS, and Flash is rewritten so that the client's web browser only retrieves content from the enterprise web application via the BIG-IP APM virtual server.

Features

BIG-IP APM has two primary access modes by which it can provide clientless access to internal web resources: Portal Access and web access management (also called LTM-APM).

LTM-APM does not rewrite the page content, and if links or other functionality reside on a different internal host, additional BIG-IP APM-protected virtual servers must be configured to support each. Additionally, the BIG-IP APM session cookie may be shared between any number of other host names in the same domain.

Portal access rewrites page content. HTML, Java, JavaScript, CSS, Flash, and other page functionality are directed through a virtual server protected by BIG-IP APM. Therefore, it does not require the additional virtual servers.

Important Rewriting complex JavaScript content may cause web page functions to malfunction. Allow for extra testing time if using portal access.

Components

The following components must be created in order to use portal access:

- A rewrite profile.
- A server SSL profile, when using internal pages protected by HTTPS.

- A portal access webtop or full webtop.
- A connectivity profile.
- An access policy that assigns a webtop and portal resource.

Implementation

The figure below shows the packet flow for a user request of a web page served by portal access.

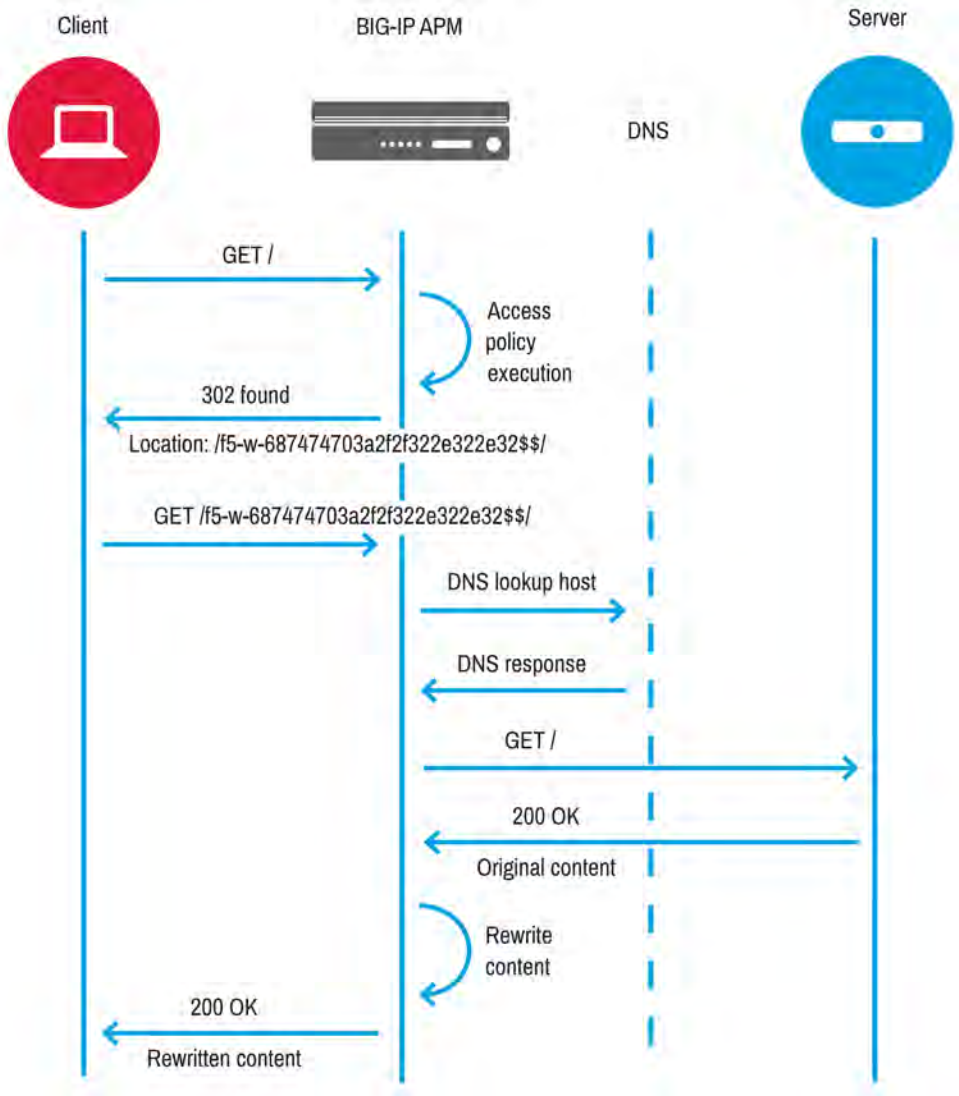

Figure 3.8: Portal access packet flow

1. User browses to a virtual server URL and initiates an access policy execution sequence.
2. After successful access policy execution, a valid session is created and portal access resources are assigned.
3. Clients access a special URL in the following format:
   https://apm/f5-w-<hex encoded scheme,host,port>$$/path.
   The URL typically comes from a portal access webtop or full webtop link, but it can also come from an iRule or from other sources.
4. BIG-IP APM validates the session, retrieves the content through the main BIG-IP APM virtual server, rewrites the content, and then returns the rewritten response.

Delivery methods for web apps

Most methods of web application delivery are supported with rewrite. However, web apps that contain JavaScript errorsor which reliy on XML stylesheets are not supported.

Reverse proxy technology is not formally standardized and new features in JavaScript libraries develop rapidly. Therefore, compatibility problems do occur in a small number of of web apps. F5 continually works to improve portal access and encourages users to report issues to F5 support.

For more information on troubleshooting, see *Troubleshooting* in this guide. For more information on communicating with F5 technical support, see *Optimize the Support Experience* in this guide.

Security considerations

There are several important security considerations regarding portal access:

- **Cookie proxy** ensures that cookies are not stored on the client browser.
- **Java rewriting** allows Java applets to access URLs and TCP sockets through BIG-IP APM.
- **ACLs are allowed.**
  - Each portal access resource item assigned to a user is considered to be an "Allow" ACL.
  - Configure ACLs to allow access only to required resources.
  - F5 recommends defining ACLs as narrowly as possible.
  - Portal access "Allow" resource items follow the same ACL priority system as other assigned ACLs.
  - If a "default deny" stance is required, an ACL with a "Deny All" entry should be configured.
    For more information, see the ACL use case in this guide.

- **Split tunnel** allows selective rewriting of the target hosts.
  - Links within rewritten pages that are included in the bypass list of the rewrite profile will not be rewritten. They are accessed directly. For example, on a page with an internet video link in an iframe, the video can bypass rewrite and the remainder of the page is rewritten.
  - For security considerations, applications that are outside of your control should not be used via rewrite. Disable this option only for testing or troubleshooting.

Citrix integration

When integrated with Citrix, BIG-IP APM performs authentication to control access to Citrix-published applications and remote desktops. SmartAccess filters can also be used.

BIG-IP APM supports the following types of integration with Citrix:

- Integration with Web Interface sites

  BIG-IP APM load balances and authenticates access to Web Interface sites, providing SmartAccess conditions based on endpoint inspection of clients. Web Interface sites communicate with XML Brokers, render the user interface, and display the applications to the client.
- Integration with XML brokers

  BIG-IP APM does not need a Web Interface site in this type of integration. BIG-IP APM load-balances and authenticates access to XML Brokers, providing SmartAccess conditions based on endpoint inspection of clients. BIG-IP APM communicates with XML brokers, renders the user interface, and displays the applications to the client.

Features

BIG-IP APM Citrix integration can simplifies a Citrix environment by replacing some of its core services, including the following:

- **Citrix Web Interface server**. In the absence of the Web Interface, BIG-IP APM can communicate with the Citrix XML Broker directly and display the user's published applications along with other non-Citrix resources.

- **Citrix Access Gateway and Netscaler.** BIG-IP APM provides a single secure entry point for both web and ICA communications and fully supports the Citrix gateway protocol functionality.

- **Citrix Secure Ticketing Authority service.** BIG-IP APM secure session token provides comparable functionality to Citrix's STA service.

Components

The different components involved in the BIG-IP APM Citrix integration depend on the mode deployed and which Citrix services are being replaced.

Integration with Web Interface sites

The following figure shows BIG-IP APM deployed as an authentication proxy for SSO on Citrix Web Interface. BIG-IP APM authenticates the client and then performs server-side SSO to the Web Interface.

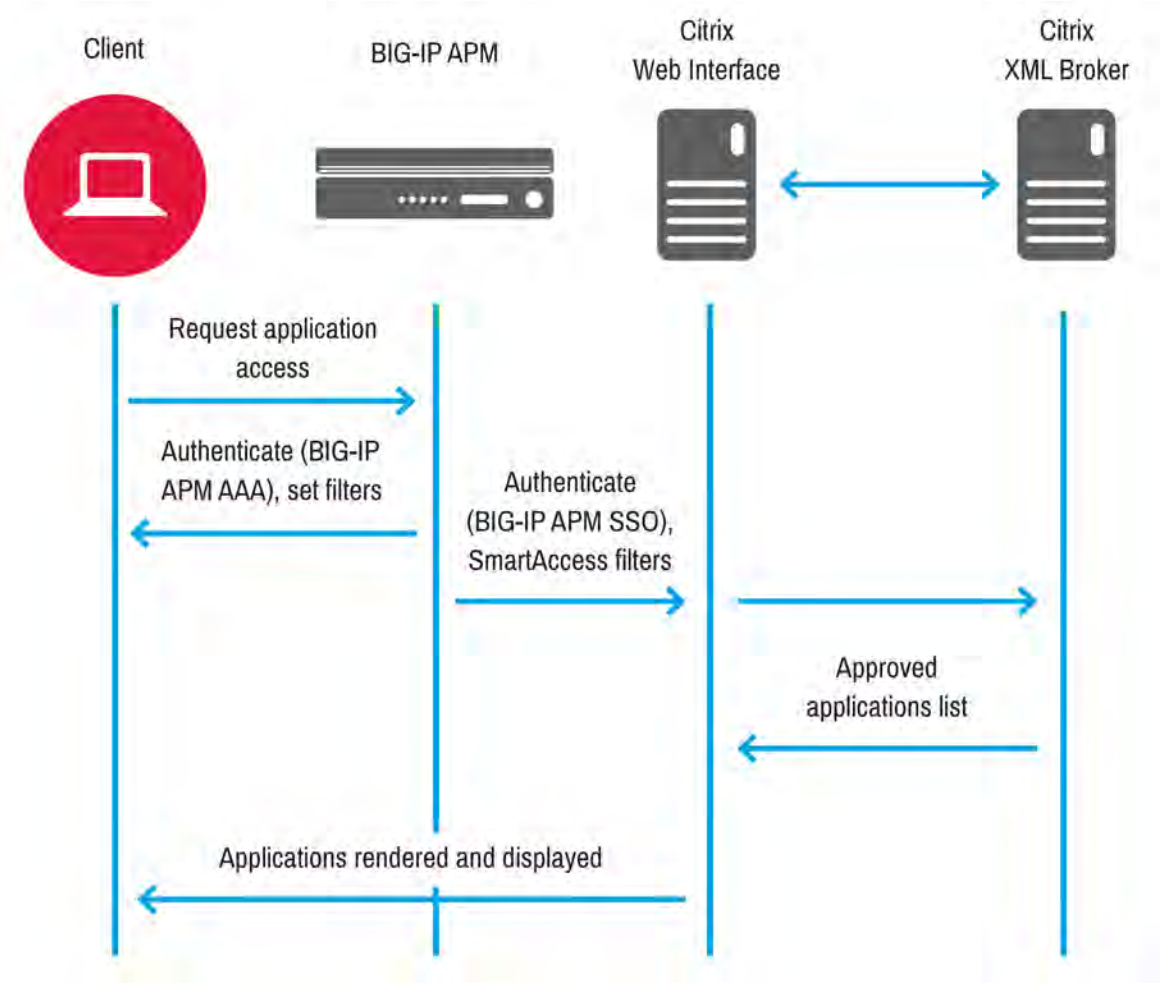

Figure 3.9: BIG-IP APM as authentication proxy for SSO on Citrix Web Interface

The following BIG-IP APM components must be created for this implementation:

- An access policy.
- An SSO profile or a VDI profile applied to a virtual server.

For more information on configuring the BIG-IP APM integration with Citrix Web Interface, see ***Integrating BIG-IP APM with a Citrix Web Interface Site***

Integration with XML Brokers

The following figure shows BIG-IP APM integration with Citrix XML broker. BIG-IP APM communicates directly with the Citrix XML Broker and displays the user's published applications and remote desktops on a common BIG-IP APM webtop.

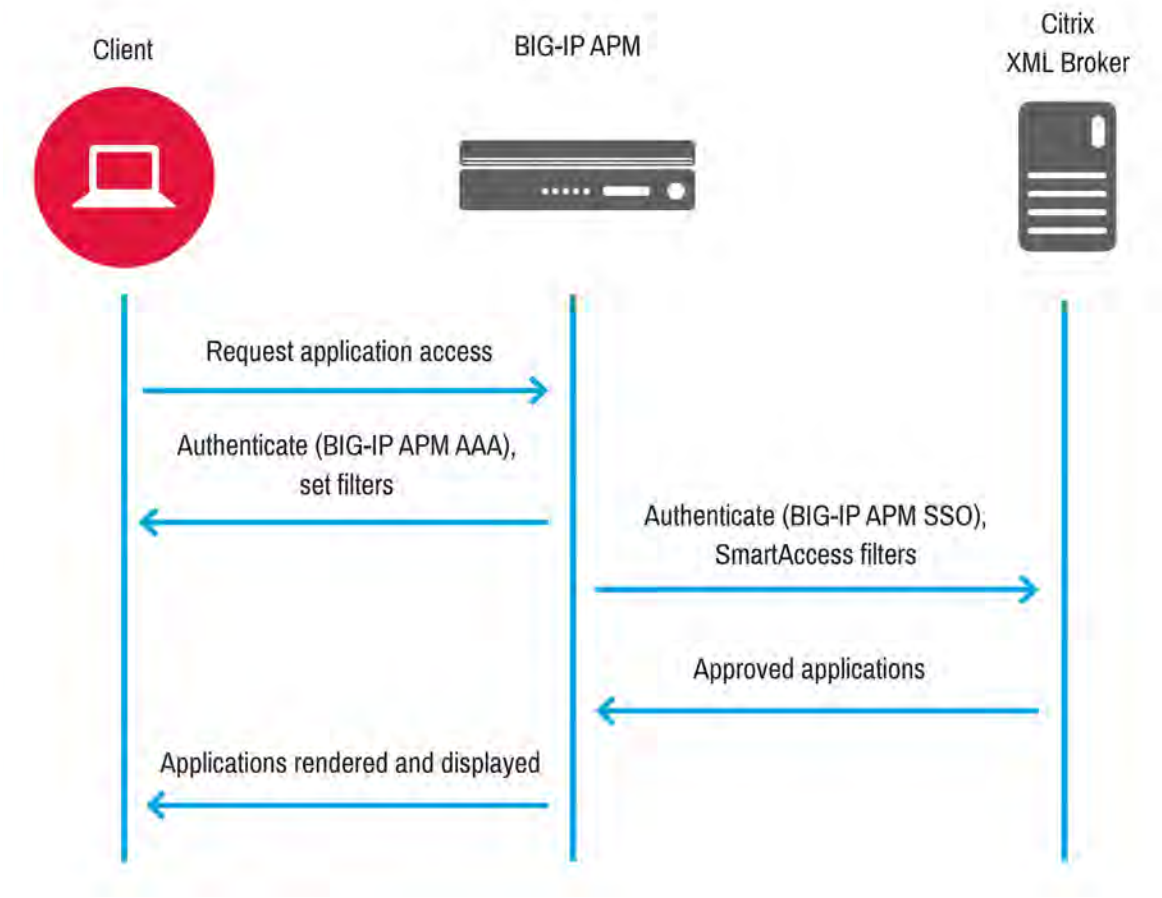

Figure 3.10: BIG-IP APM integration with Citrix XML broker

The following BIG-IP APM components must be created for this implementation:

- An access policy
- A webtop
- A Citrix Remote Desktop resource
- A connectivity profile

For more information on configuring the BIG-IP APM integration with XML Brokers, see ***Integrating BIG-IP APM with Citrix XML Brokers***

Using an iApp

An F5-supported iApp is available to help simplify BIG-IP APM Citrix integration by configuring the required settings for this deployment. The iApp can be found in the ***DevCentral iApp CodeShare*** .

Note A DevCentral login is required to view this content.

VMware View support

When integrated with VMware View, BIG-IP APM performs authentication to control access to published View remote desktops and optionally simplifies the View environment by replacing the VMware Security Server.

BIG-IP authenticates standalone VMwareView clients.

- Performs authentication and load balance VMware View connection servers.
- Supports the PC-over-IP (PCoIP) display protocol for the virtual desktop.
- Allows a View client to make connections to support different types of traffic between it and a View connection server.
- Supports View client connections with two virtual servers—one TCP port 443 and one UDP port 4172—which share the same destination IP address.

BIG-IP presents VMware View desktop on a webtop.

- Integrates with View connection servers to present View desktop resources on a BIG-IP APM dynamic webtop.
- Authenticates to a View connection server and renders the View desktop resources.
- Load balances the View connection servers for high availability.
- Supports the necessary connections with two virtual servers that share the same destination IP address.

Features

BIG-IP APM VMware View integration can enhance a View environment in the following ways:

- **Replaces the View security server service.** BIG-IP APM functions as a native PCoIP proxy that provides a single secure entry point into the View environment.

- **Allows webtop integration.** BIG-IP APM renders VMware View remote desktop resources within a common webtop that can include other non-View resources.

- **Allows layered security.** BIG-IP APM can enhance the PCoIP communications with a layer of DTLS (TLS for UDP).

- **Employs single namespace and user persistence.** By adding BIG-IP GTM, BIG-IP APM can provide a single common namespace for all View pods. By querying a user directory, BIG-IP APM can persist a user to a selected datacenter.

Components

The following BIG-IP APM components must be created for VMware View support:

- A VMware View remote desktop resource.
- A full webtop.
- An access policy.
- A connectivity profile.

For more information on configuring authenticated standalone View client access, see *Authenticating Standalone View Clients with APM* in ***BIG-IP Access Policy Manager: Third-Party Integration Implementations.***

For more information on configuring webtop presentation of View remote desktop resources, see *Presenting a View Desktop on an APM Webtop* in ***BIG-IP Access Policy Manager: Third-Party Integration Implementations***.

iApp deployment

For iApp deployment options that are supported by F5, see VMware applications at *DevCentral iApp CodeShare* The listed iApp will configure all of the required settings for both deployment options discussed in this section.

Remote Desktop Protocol support

BIG-IP APM Remote Desktop Protocol (RDP) provides secure access to internal Microsoft Remote Desktop Services.

Multiple deployment options can be used to enable the preferred end-user experience.

Features

Providing access via a proxied method can be faster and less troublesome than via a full VPN connection. Flexible authentication options mean that RDP resources can be protected by any type of access policy.

ACLs can be used to enforce network use policies. For more information, see *Access Control Lists* in this chapter.

Components

The following BIG-IP APM components must be created for RPD support:

- A connectivity profile.
- An application tunnel profile for an RD resource with Java client enabled (Java & Per-App VPN).

Implementation

Remote Desktop resource

- A user connects using web browser, authenticates, and selects a preconfigured Remote Desktop resource from a full webtop.
- A browser launches a new window that contains either MS RDP Web ActiveX or a Java Applet to provide client services through APM.

Note The Remote Desktop resource target can be based on a dynamic session variable.

Application Tunnel for mstsc.exe

- A user connects using a web browser and selects an application tunnel resource from a full webtop.
- A user launches **mstsc.exe** to connect to internal server through a standard application tunnel.

This option has the most client flexibility and familiar user experience, but lacks SSO capability.

RD Gateway for clients

- A user launches a Remote Desktop Connection client and connects directly to the BIG-IP APM virtual server.

For more information, see *Overview: Configuring APM as a gateway for Microsoft RDP clients* in ***BIG-IP Access Policy Manager: Application Access Guide***.

Exchange proxy

Exchange proxy is the F5 BIG-IP APM solution to provide secure remote access for all Microsoft Exchange services. These include:

- ActiveSync
- Autodiscover
- Exchange Web Services
- Offline Address Book
- Outlook Anywhere
- Outlook Web Access

Features

Exchange proxy provides NTLM authentication functionality for Microsoft Exchange clients, such as Microsoft Outlook, iOS Mail, and Android email. The solution is provided in such a way that it can be used simultaneously with multiple client-side authentication types (HTTP Basic, HTTP NTLM, and more) and authentication for mobile devices, depending on capability and protocol used.

The solution identifies the remote clients and forces them to successfully authenticate before forwarding requests to the Exchange Client Access Server (CAS) service. This process provides enhanced security and auditing capability.

If configured, the solution can also provide SSO functionality for Exchange services.

Components

The following components must be created in order to use the Exchange proxy:

- An NTLM machine account.
- An NTLM authentication configuration.
- An Exchange profile.
- A Kerberos SSO profile.
- Support for HTTP Basic for Autodiscover/ActiveSync.
- An access profile with an Exchange profile assigned.

Implementation

The following figure shows the packet flow between BIG-IP APM and the client, BIG-IP APM and the Active Directory Server, and BIG-IP APM and the Exchange CAS service.

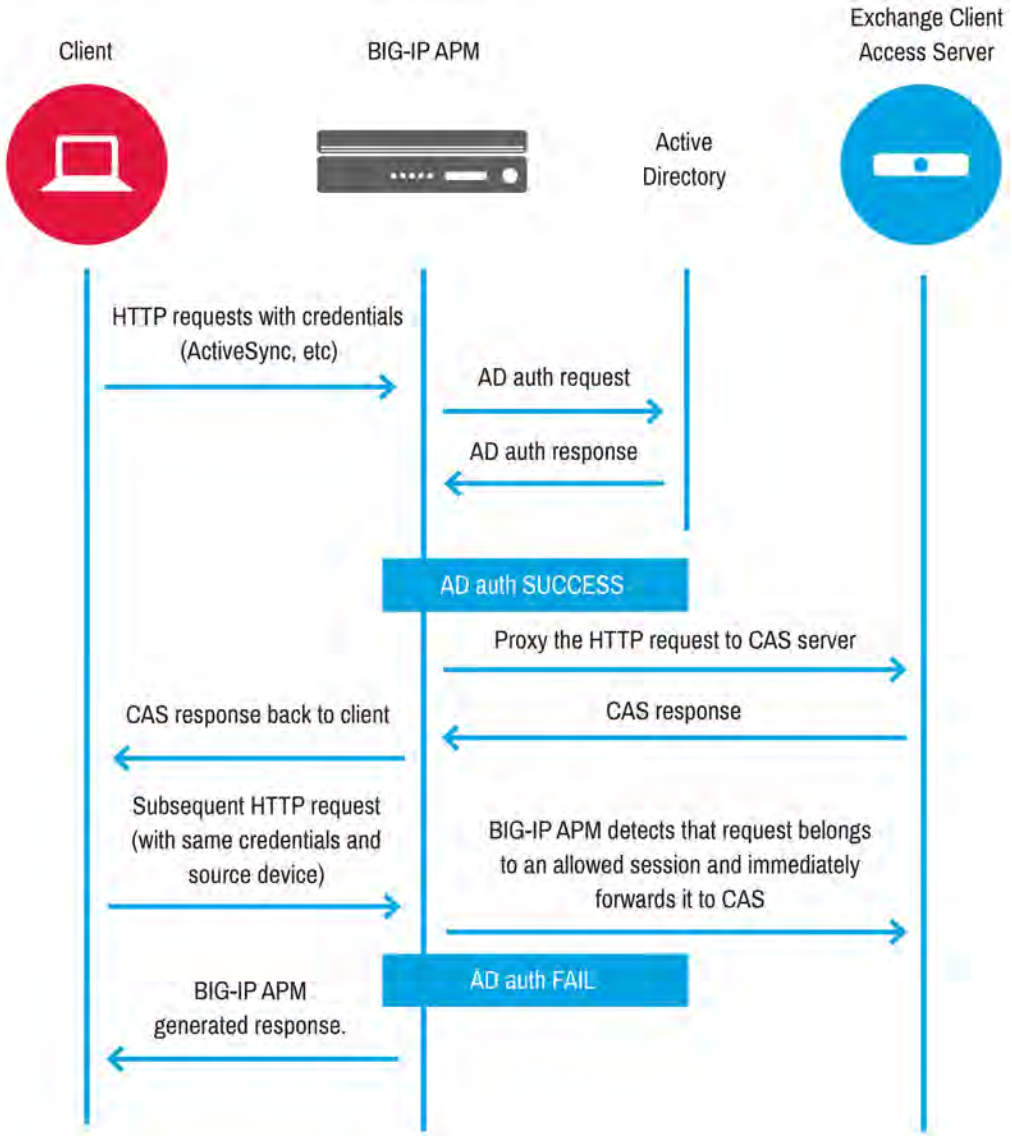

Figure 3.11: Exchange proxy packet flow

For more information, see *HTTP Basic Authentication for Microsoft Exchange Clients* in ***BIG-IP APM Authentication Configuration Guide*** and *Exchange Deployment Guide*.

Webtop

A webtop is a BIG-IP APM customizable landing page. At the end of successful access policy execution and final client POST to complete the access policy, the client can be redirected to a BIG-IP APM webtop.

Webtop types

BIG-IP APM supports three types of webtop:

- **Network access.** Contains Javascript and browser plugins to launch network access on supported browsers or the BIG-IP Edge Client.
- **Portal access.** Contains a 302 redirect to the portal access encoded URL.
- **Full webtop.** Contains a complex set of Javascript, XML, and HTML to present a menu to end-users. Assigned resources are presented to the user as icons. A full webtop also allows the launching of network access from a browser and the BIG-IP Edge Client.

Note If no webtop is assigned during access policy execution, the session is in web access/LTM-APM mode.

Features

The full webtop can replace intranet or extranet portal pages, offering users a centralized place to launch assigned applications.

Network access and portal access webtops automatically place users into a specific application assigned during access policy execution.

BIG-IP APM provides a basic customization framework allowing administrators to alter images, color, and layout settings.

The advanced customization framework allows web developers to completely replace all BIG-IP APM-delivered web content, including webtops, logon pages, and error pages.

Implementation

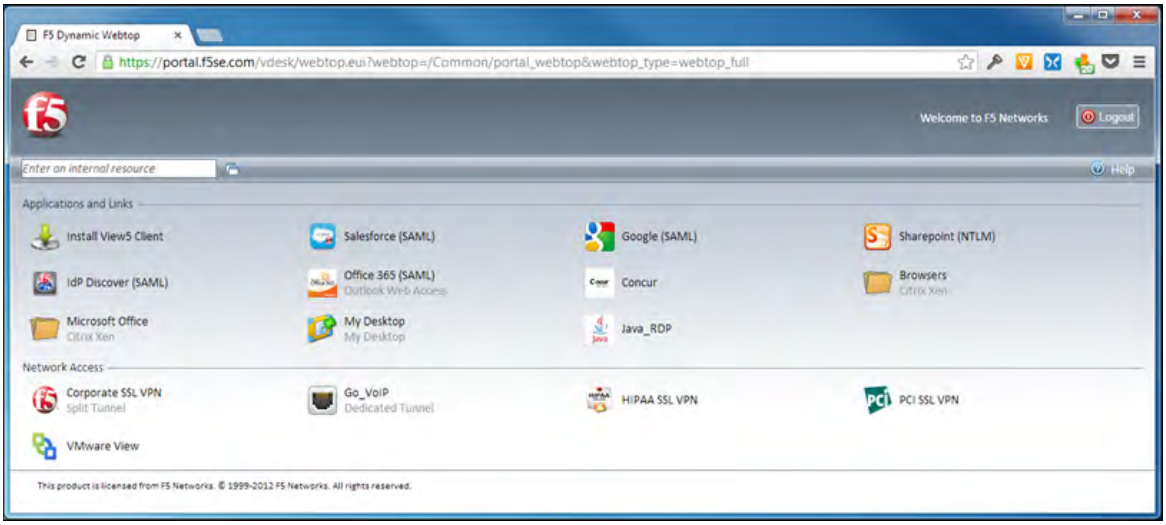

Figure 3.12: Sample BIG-IP APM full webtop

Access control lists

Access control lists (ACLs) are used to restrict user access to specified internal hosts, ports and/or URIs. For an ACL to have an effect on traffic, it must be assigned to a user session. ACLs are applied to all access methods by default.

An ACL consists of a list of access control entries (ACEs). These entries can work on layer 4, layer 7, or both.

In addition to source (ip:port), destination (ip:port), and Scheme + URI (for L7), each ACL and its entries has a unique "acl-order" field that determines its priority.

Important: Certain types of resource items are considered to be "allow" ACLs. These include portal access, app tunnels, and remote desktop.

A user session is assigned a list of ACLs during access policy execution. BIG-IP APM tests ACLs and ACEs in order, based on their priority in the respective list. To make sure of compliance with network use policies, the order must be correct.

If there are no ACLs assigned to a session by the access policy, the default behavior for the session traffic is "allow."

If a "default deny" stance is required, an ACL with a "Deny All" entry should be configured. This ACL should be assigned to the user session at the end of the ACL entry list (that is, its order field value should be highest number). BIG-IP APM will then reject any connection not matched by a previous entry.

ACLs can be configured to create log entries when they are matched. These log entries appear in the **/var/log/pktfilter** log file. They can also be viewed in the **Configuration utility**. Go to **System** > **Logs** > **Packet Filter**.

When an ACL is applied to a BIG-IP APM access policy, the access policy dynamically creates an internal layered virtual server that is used to apply the ACL. However, if the

BIG-IP APM virtual server targets a layered virtual server, such as an SSO layered virtual server, traffic bypasses the dynamically-created internal layered virtual server and the ACL is not applied.

For more information, see AskF5 article ***SOL14219: An L4 ACL has no effect when a layered virtual server is used*.**

Dynamic ACLs

A dynamic ACL is an ACL that is created on and stored in an LDAP, RADIUS, or Active Directory server. A dynamic ACL action dynamically creates ACLs based on attributes from the AAA server. Because a dynamic ACL is associated with a user directory, you can use it to assign ACLs specifically per the user session. BIG-IP APM supports dynamic ACLs in an F5 ACL format, and in a subset of the Cisco ACL format.

When using dynamic ACLs, make sure of the following:

- The dynamic ACL appears after authentication in an access policy since its actions are determined by attributes received from an authentication server.
- If configured in a Cisco format, the dynamic ACL contains the prefix **ip:inacl#**.

For more information, see *Configuring Dynamic ACLs* in ***BIG-IP Access Policy Manager: Implementations***.

BIG-IP Edge Client

Introduction

Client types

BIG-IP Edge Client components

Client delivery

Introduction

BIG-IP APM can provide secure remote access to system resources using a number of different clients depending on deployment and user requirements. For more information, see the following:

- ***BIG-IP Client Compatibility Matrix***
- ***BIG-IP Edge Apps Client Compatibility Matrix***
- Ask F5 article: ***SOL15326:*** *Browser plugin support for BIG-IP APM features*

Client types

There are several different BIG-IP APM clients that are capable of interacting with BIG-IP APM.

BIG-IP Edge client

BIG-IP Edge Client is a native platform-specific application for desktop operating systems that provides network access and endpoint inspection. It can also launch third-party applications configured in access policy on BIG-IP APM. BIG-IP Edge Client cannot provide portal access.

For more information, see *BIG-IP Windows edge client* and *BIG-IP MAC edge client*

Mobile clients

F5 supports BIG-IP Edge Client and BIG-IP Edge Portal on mobile operating systems like Apple iOS and Android. These applications can be downloaded from their respective vendor stores. Windows Phones 8.1 and higher have BIG-IP Edge Client built-in (called Edge Client).

BIG-IP Edge Client allows users to connect to the BIG-IP APM system to provide layer 3 network access to protected enterprise network resources.

BIG-IP Edge Portal provides secure remote access to protected enterprise web applications. It can be used to bookmark frequently visited web applications to automatically authenticate. It also provides access to web applications using the BIG-IP APM content rewrite proxy.

For more information, see ***Inbox F5 VPN client*** and ***BIG-IP Edge Apps***

BIG-IP Edge command line interface clients

The BIG-IP Edge command line interface (CLI) clients do not have a graphic user interface. They are currently supported for desktop operating systems to provide network access, exclusively. They provide basic multi-factor authentication with client-certificates and username/password.

CLI clients run in "legacy-logon mode" and cannot render any HTML content. Therefore, certain access policy action items requiring client-side interaction, such as message boxes and endpoint inspection, cannot be used with the CLI clients.

For more information, see *BIG-IP Edge command line client*

BIG-IP APM Edge Client SDK

BIG-IP APM Edge Client provides an SDK that can be integrated with third-party applications. These can provide customized SSL-VPN applications capable of establishing network access with BIG-IP APM. The SDK is only supported on computers running Windows.

For more information, see *BIG-IP APM Edge Client SDK*.

Browsers

BIG-IP APM provides support for popular browsers such as Internet Explorer, Firefox, Chrome, and others. Browsers can create application tunnels using either Java applet or browser ActiveX controls in addition to providing portal access to web applications.

For more information about browsers and operating system platforms, see *BIG-IP APM client compatibility matrix guide.*

BIG-IP Edge Client components

BIG-IP Edge Client components can be installed on Windows desktop operating systems as a service or as a standalone binary.

The components broker services including:

- DNS relay proxy
- Machine certificate checks
- Recurrent endpoint posture checking
- Layer 3 tunneling
- Components Installation
- COM interface

For more information, see the following documents:

- *Windows components*
- *MAC components*
- *Linux components*

Client delivery

The BIG-IP Edge Client installation package can be delivered to users through regular file sharing methods. Administrators can also host it on the content repository hosted on BIG-IP APM. From there, the package can then be downloaded by configuring a webtop portal link to it.

For more information about BIG-IP Edge Client customization, see *Hosting a BIG-IP Edge Client Download with Access Policy Manager* in ***BIG-IP Access Policy Manager: Hosted Content Implementations***.

Customization

BIG-IP Edge Client can be customized before being delivered to users. Existing client customizations are retained following an upgrade, but administrators can create a new package with different customizations to push to users if new customizations are needed.

For the languages BIG-IP APM supports, the following BIG-IP Edge Client settings can be customized:

- Banner background color
- Banner text color
- Logo
- Tray Icon set
- About links
- About text
- Application name

For more information, see *Personalizing Client Appearance in General View* in ***BIG-IP Access Policy Manager: Customizaton***.

Uninstall

For information regarding uninstalling client components, see AskF5 articles ***SOL8253: Removing BIG-IP APM and FirePass client components from Windows client systems*** and ***SOL15539: Removing the BIG-IP Edge Client from Mac OS X devices*****.**

To uninstall BIG-IP Edge Client and components on computers running Apple OS

- Go to **BIG-IP Edge Client Menu > Preferences > Advanced > Uninstall**.

Caution: Before uninstalling BIG-IP Edge Client, first uninstall any components that have been installed separately from the BIG-IP Edge Client.

Upgrade Behavior

When BIG-IP APM system software is upgraded to a newer version, rollup hotfix, or engineering hotfix, the BIG-IP Edge Client may auto-upgrade depending on the **Component upgrade** setting in the **Connectivity profile**.

To change the auto upgrade setting in the BIG-IP Edge Client component using the Configuration utility

1. Go to **Access Policy** > **Secure Connectivity**.
2. Click the **Connectivity profile** name and then click **Edit Profile**.
3. Click **Win/Mac Edge Client**.
4. Change the setting in the **Component Update** menu.

By design, some BIG-IP Edge Client components will be updated regardless of how this setting is configured. These components include:

- Endpoint inspection components.
- Policyserver (Apple OS only).
- Network access plugin (Apple OS only).
- SAM Inspection host plugin (Apple OS only).
- TunnelServer.
- VPN for Windows.
- SuperHost for Windows.
- Host for Windows.
- InspectionHost for Windows.

- InstallerControl for Windows.

Note Even if **Component Update** is set to **Yes,** users can still manually cancel the upgrade and BIG-IP Edge Client will continue with its old version.

Security

Introduction

Session management

Identity and access management

Network security

Auditing

Introduction

BIG-IP APM provides security through session management, session ID rotation, identity access management, tunneling, ACLs, and several other measures.

Session management

BIG-IP APM security is founded on how BIG-IP APM session managment: that is, how sessions function and terminate. Several security protections function at the session level:

- Session ID rotation provides protection against session-level hacking.
- Maximum sessions and timeouts can be configured.
- Cookie options can be configured.
- Access policies can be configured based on the client IP address' reputation.
- Back-end authentication servers protection from distributed denial of service (DDoS) attacks and users protection from lockout due to such attacks.

Session ID rotation

All BIG-IP APM client sessions are tracked using a unique, proprietary session ID. During the course of an access policy evaluation, the session ID is randomly rotated to prevent session hijacking and fixation attempts.

Session ID rotation is enabled by default in versions 11.2.0 and higher. This feature can cause issues with older clients or deployments using iRules if they rely upon remembering Session ID.

To disable automatic Session ID Rotation using tmsh from the command line

1. To view the current configuration setting for session ID rotation, type the following command:

```
tmsh list /sys db apm.rotatesessionid
```

2. Disable automatic rotation of BIG-IP APM session IDs by typing the following command:

```
tmsh modify /sys db apm.rotatesessionid value disable
```

3. Save the change by typing the following command:

```
tmsh save /sys config
```

Maximum sessions per user

For some access profile types, administrators can set a custom value for the maximum number of valid sessions a user can have open at one time.

Since each session consumes an access license, malicious attacks to consume all BIG-IP APM licenses will fail.

To set a custom value for maximum sessions per user using the Configuration utility

1. Go to **Access Policy** > **Access Profiles: Access Profiles List**.
2. Click a profile name.
3. Select **Custom**.
4. Under **Settings,** set a value for **Max Sessions Per User**.
5. Click **Update**.

Session timeouts

The timeout values of all BIG-IP APM sessions are controlled in the definition of an access profile. The values can be used to expire sessions based on inactivity, maximum session time, or exceeding of access policy evaluation time.

Once the inactivity timeout setting is configured, there are only four events or actions that override it:

- The user logs out of BIG-IP APM.
- The value for **Maximum Session Timeout** is exceeded.
- A BIG-IP APM administrator deletes the session.
- The access policy evaluation exceeds the configured timeout period.

Secure cookies

To make sure that the client browser will not send session cookies in an unencrypted request, the **Secure cookie** option (enabled by default) adds the secure attribute to the session cookie.

The following shows an example of a session cookie with **Secure cookie** enabled:

```
Set-Cookie: MRHSession=d896020385383db9ece7ac6d41f45923; path=/; secure
```

Note Because the secure cookie option makes sure the browser will not send a session cookie in an unencrytped request, the secure cookie option needs to be disabled in application access control deployments where an HTTP virtual server is used.

HTTPOnly cookies

For browsers that support it, the HTTPOnly option can be enabled to mitigate the risk of a client side script accessing the BIG-IP APM session cookies. This option only works for the LTM+APM access profile type. Other access profile types require access to various session cookies.

Persistent cookies

Persistent cookies can be used with web access management/LTM-APM access profile type to store the cookies locally on the client hard disk. When the session is first established, BIG-IP APM session cookies are not marked as persistent. After the user successfully authenticates with BIG-IP APM and the access policy completes successfully, the cookies will be marked as persistent in the next response to the client.

Network access and application tunnels do not support persistent session cookies.

Restriction of sessions to a single client IP

User access to BIG-IP APM can be restricted to a single client IP address via configuration in the access profile. Enabling the **Restrict to Single Client IP** option will associate a client's IP address to their BIG-IP APM session. Upon each client request, BIG-IP APM will verify that the client's IP address associated with the BIG-IP APM session has not changed.

If the IP address has changed, the session is terminated, the request is redirected to the Access Profile's logout page and a log entry is written to **/var/log/apm** indicating that a session hijacking attempt was detected.

This setting may not be very useful in deployments where users hop between Wireless Access Points, as these access points could give different IP numbers to genuine users.

DoS/DDoS protection

The **Max In Progress Sessions Per Client IP** restricts the number of in-progress access policies for a given client IP address. In-progress access policies are client sessions which are still being evaluated by BIG-IP APM to determine whether the client will be granted or denied access to the protected resources.

The default value only allows 128 in-progress sessions per IP. After reaching that value, the BIG-IP APM will deny any additional requests from that IP to start a new BIG-IP APM session. Versions prior to 11.6.0 used a default value of 0. A value of "0" represents unlimited sessions. If the value is less than 128, consider increasing it.

Caution: If a large number of users simultaneously access BIG-IP APM from behind a device or proxy configured with NAT it may prevent new sessions from starting.

Brute-force protection

The **Minimum Authentication Failure Delay** and **Maximum Authentication Failure Delay** options can be enabled to slow down or mitigate brute-force attacks against BIG-IP APM.

The following AAA types can be used:

- Active Directory
- HTTP
- Kerberos
- LDAP
- Local User DB
- One-time password verification
- Oracle Access Manager
- RADIUS
- SecurID
- TACACS+

Another option to prevent brute-force attacks is to use CAPTCHA on a BIG-IP APM logon page. By default, BIG-IP APM uses the Google reCAPTCHA service, but any CAPTCHA service that provides a reCAPTCHA-compatible API can be used.

The Local User DB also provides a user account lockout option. After a certain amount of failed login attempts, the user account is locked out for the specified interval. Lockout limitations include the following:

- An attacker can cause a DoS/DDoS by locking out large numbers of accounts.
- Lockout is ineffective against slow attacks that try only a few passwords every hour.
- If an attack continues after one or more accounts are locked out, system resources will be consumed.
- Lockout can reveal information about valid users to attackers during a directory harvest attack.

For more information on how Local User DB can be used to mitigate brute force authentication attacks, see *Local User Database* in ***BIG-IP Access Policy Manager: Authentication and Single Sign-On***.

Identity access management

Identity and access management (IAM) consists of the management of user authentication, authorization, and privileges within the enterprise and cloud-based services. The goal of IAM is to increase security and productivity while decreasing cost, downtime, and repetitive tasks.

As many organizations move to adopt more cloud-based services, they may experience challenges in ensuring that ACLs are accurate across all services. They may also experience difficulty enforcing a reliable security policy.

As with internally managed services, Software-as-a-Service (SaaS) providers maintain their own IAM systems for usernames, passwords, and access control enforcement. This siloed approach can result in security management issues as organizations employ multiple, unintegrated IAM systems.

BIG-IP APM access federation addresses these issues by eliminating the disconnect between internally maintained IAM systems and services external to the enterprise. In doing so, BIG-IP APM can deliver consistent security everywhere.

Password management is simplified by maintaining all user passwords in the corporate user directory. Users require only a single password to access internal systems and SaaS-based applications.

Multifactor authentication can be implemented at the access control point (BIG-IP APM). Client-side inspection checks (such as firewall and antivirus systems and machine certificate check) can be performed before allowing access to applications.

Multi-factor client authentication

The three most common authentication factors are known as "Something You Know," "Something You Have," and "Something You Are."

- "Something You Know" is typically a password or a PIN, but may include questions that only the user can answer, or a touchpad gesture.
- "Something You Have" is typically either a physical or software token, but might include a certificate, or some combination of certificate and token.
- "Something You Are" is typically a biometric input, such as fingerprint, retina scan, or facial or voice recognition.

Multi-factor client authentication includes a combination of two or more authentication factors.

Client credentials protection

All in-memory sensitive data, such as user credentials, SSO credentials, and secure stored session variables, are 128-bit AES encrypted. BIG-IP APM uses a per-user master key, which is derived from the BIG-IP APM session cookie. The cookie is only valid for that single user session. The key is not stored in memory, and the session variables are only stored in memory as long as the session is valid. Once the session is terminated, the data is removed and the key is destroyed.

Network security

Two of the configurable network security features BIG-IP APM provides are tunnels and ACLs.

Tunnelling

BIG-IP APM offers support for either full or split tunneling for network access.

- Full tunnelling provides Windows, Macintosh, Linux, and Windows Mobile users with access to the complete set of IP-based applications, network resources, and intranet files available, as if they were working at their desktop in the office.
- Split tunnelling provides control over exactly what traffic is sent over the network access connection to the internal network and which is not. This feature provides better client application performance by allowing connections to the public Internet to go directly to the destination, rather than being routed down the tunnel and then out to the public Internet.

Full tunnelling

Of the two types of tunnels, full provides better security for clients. In a full-tunnel deployment, all VPN traffic is sent through the VPN tunnel, which allows for greater control of traffic from remote users. Traffic destined for the Internet can now traverse through the company's gateway security devices and have corporate policy applied to it.

As more and more enterprises elect to pay for the extra resources required to maintain VPN full tunnels, the concern becomes the sites to which users are connecting when the devices are not connected to the VPN. This has led to many companies allowing only corporate-issued devices to be connected to the VPN, and a requirement that those devices must always be connected to the VPN. This forced-VPN connection is deemed by many to be the most effective method of monitoring an offsite corporate asset.

Split tunnelling

Split tunneling results in less traffic flowing through BIG-IP APM, as only traffic destined for the VPN traverses the tunnel. Less traffic leads to a smaller workload for BIG-IP APM and lowered bandwidth requirements due to less VPN traffic. Split tunneling also allows for a strict separation between corporate intranet traffic and private Internet use.

Split tunneling also allows a user to access more than one network without having to repeatedly connect and disconnect in order to switch from one network to the other.

Security concerns with split tunnelling

Split tunneling makes it possible for a remote user to bypass network security, such as web content filtering for outbound HTTP traffic. If users are connected to a remote network employing DNS hijacking, name resolution may not work as expected for internal hostnames. To mitigate this issue, BIG-IP APM provides Windows DNS Relay Proxy, which can be configured to allow hostnames for internal domain names to be intercepted and relayed over the VPN tunnel for correct resolution.

For more information, see AskF5 article: ***SOL9694: Overview of the Windows DNS Relay Proxy service***.

Another security concern is that split tunneling can allow a PC infected with malware to act as a gateway to the corporate network. To protect split tunneling configurations against such exploitation, BIG-IP APM provides two options that can be used to further protect split-tunnel configurations: Prohibit routing table changes during Network Access Connection and Integrated IP filtering engine.

When enabled, the **Prohibit routing table changes during Network Access connection** option prevents the client's routing table for the F5 Point-to-Point Protocol (PPP) adapter from being changed by adding, deleting, or modifying any existing routes. The adapter is constantly monitored by the F5 VPN software. If any changes to the routing table are found, they are discarded and the routing table reset.

When enabled, the **Integrated IP filtering engine** provides additional security for data leakage. Traffic generated by devices on the client's LAN are not allowed to traverse the tunnel.

ACLs

Once a BIG-IP APM tunnel is established on a network, users with access to the tunnel have full access to the back-end network. To improve security, remote user access can be limited to crucial network resources only with ACLs. ACLs can be configured on a per-session basis to provide individually tailored access for each user.

BIG-IP APM uses both static pre-configured ACLs and dynamic ACLs. Dynamic ACLs live on other devices, such as Active Directory or RADIUS. Many companies will configure ACLs on their internal routing and switching infrastructure. Matching on the source address, which is the tunnel address, provides defense against potential network access violations.

Auditing

By enabling audit logging, BIG-IP APM can track configuration changes. When audit logging is enabled, commands performed by an administrator and by root are logged to the **/var/log/audit** file. These includes changes to access policy using the Visual Policy Editor and creation, deletion or modification to any profile, AAA server, resource, or other objects. Changes are logged as **tmsh** commands regardless of whether the logged action was performed in the **Configuration utility** or at the command line.

High availability

Introduction

Failover components

High availability

Policy sync

High availability on VIPRION

Introduction

A high availability (HA) pair of BIG-IP APMs indicates a redundant system deployment consisting of two BIG-IP systems synchronized with the same configuration. One system actively processes traffic while the other remains in standby mode until needed.

The goal of such redundant pairing is to provide users with seamless, uninterrupted service in the event of failure on one device.

If the active system is taken offline or something occurs to prevent it from processing traffic the standby system immediately begins processing traffic. Typically, the newly active system remains active until an event requires the first BIG-IP system to become active again or until an administrator manually forces that system into standby.

While BIG-IP system configurations allow for configurations with multiple standby systems or active-active pairings, BIG-IP APM only supports two systems paired in active-standby configuration.

For more information, see AskF5 article: ***SOL15503: BIG-IP APM HA considerations***.

BIG-IP APM failover components

Synchronization of data between BIG-IP systems in a high availability configuration is made possible through device trust domains, device groups, and traffic groups.

Device trust domains

To provide failover or configuration sync, BIG-IP APM systems on the network must be in the same trust domain. The trust relationship between BIG-IP APM devices on the network is established through certificate-based authentication. BIG-IP APM devices in a trust domain can synchronize and failover their BIG-IP APM configuration data, and exchange status messages continuously. A local trust domain includes the BIG-IP system local device.

Device groups

A device group is a collection of BIG-IP systems that have established a device trust and share data with each other. The type of data shared depends on what type of data the device group is configured to share.

The two device groups types are sync-only and sync-failover.

- **A sync-only device group synchronizes only configuration data**, such as policy data, but it does not synchronize failover objects. This configuration is typically used for synchronizing configuration data between BIG-IP systems deployed in different geographic locations.

- **A sync-failover device group synchronizes configuration data and traffic group data for failover purposes.** This configuration allows for full synchronization

between two BIG-IP systems. If the active system becomes unavailable, failover occurs and the standby system is able to instantly pick up traffic passing through the system without interruption.

Traffic groups

A traffic group is a collection of related configuration objects that run on a BIG-IP system. Together, these objects process a particular type of traffic on that device. In general, a traffic group makes sure that when a device becomes unavailable, all of the failover objects in the traffic group fail over to a standby system in its device group.

High availability

In order to deploy BIG-IP APM systems in an high availability (HA) configuration, two requirements must be met:

- A device trust needs to be established between two systems.
- A sync-failover device group must be configured.

For more information, see AskF5 article: ***SOL13649 - Creating a device group using the Configuration utility***.

Configuring HA

- Active-standby deployments with BIG-IP APM should use **traffic-group-1** only.
- Only active-standby configurations between two devices are supported.
- BIG-IP APM peer devices must use identical hardware and run the same software version including hotfix level.

For more information, see AskF5 article: ***SOL8665 - BIG-IP redundant configuration hardware and software parity requirements***.

BIG-IP APM session data is synchronized across an active-standby deployment. This can be verified from the command line using the **sessiondump** utility. By identifying a newly created BIG-IP APM session on the active unit and using **sessiondump** to verify whether or not the BIG-IP APM session exists on the standby unit.

For more information about **sessiondump**, see *Troubleshooting: Tools and Utilities* in this guide. Also see AskF5 article: ***SOL11134: Locating a user's session ID*.**

Failover and user sessions

The BIG-IP system supports both persistence and connection flow mirroring. In a typical BIG-IP LTM use case, both are of these are used to allow TCP connection states to be shared between HA peers.

However, BIG-IP APM does not support the use of connection flow mirroring. BIG-IP APM HA mirroring configurations synchronize BIG-IP APM session data (such as session variables and session state). In the case of a failover event, the client session state is maintained and the user will not be required to re-establish the session.

The following items describe typical BIG-IP APM failover behavior:

- Active BIG-IP APM session is maintained.
- TCP and PPP connections are reset.
- Disconnected clients automatically renegotiate their network connection.
- User sessions are maintained, including session state information.
- Client applications re-establish connection when the tunneled protocol supports automatic reconnect (such as VPN and RDP protocols).

Note: Depending on the how they are connected, client applications such as FTP and SSH may not automatically re-establish their connections.

The following figure shows an example of a failover between APM01 and APM02, deployed in an HA pair.

Figure 6.1: BIG-IP APM failover

1. On an existing connection, client request traffic arrives at a virtual server on APM01 (active unit).

2. APM01 checks its connection table entries and processes the traffic when the entry is matched.

3. APM01 sends response traffic back to client.

4. APM01 experiences a failover event and APM02 becomes the active unit.

A gratuitous ARP updates network infrastructure ARP caches with the MAC address of the appropriate interface on APM02 for the virtual server's IP.

5. The client sends a request using its existing TCP connection and the traffic arrives at APM02 (now the active unit).

6. APM02 checks its connection table entries and does not match an entry for the client.

7. APM02 sends a TCP RST to the client.

8. Client initiates a new TCP handshake.

9. APM02 makes an entry in its connection table. A new SSL session is negotiated, and since APM session information is mirrored, the session resumes.

10. APM02 sends response traffic back to client.

Policy Sync

BIG-IP APM Policy Sync functionality allows access policies to be maintained on multiple BIG-IP APM devices. It does this while adjusting appropriate settings for objects that are specific to device locations, such as network addresses. Policies can be synchronized from one BIG-IP APM device to another BIG-IP APM device, or to multiple devices in a device group.

A sync-only device group configured for automatic and full sync is required to synchronize access policies between multiple devices.

For more information, see ***Synchronizing Access Policies*** ***in BIG-IP Access Policy Manager: Implementations***.

Note A maximum of 32 BIG-IP APM systems are supported in a sync-only group type.

High availability on VIPRION

Special considerations should be taken into account when deploying BIG-IP APM on a VIPRION system.

- BIG-IP LTM currently supports active-active deployments across traffic groups and up to 32 BIG-IP systems.
- BIG-IP APM only supports active-standby deployments across two BIG-IP APM systems.
- BIG-IP APM can only run from within traffic-group1 on a BIG-IP system.

A BIG-IP VIPRION system can be deployed in two different ways:

- Standalone
- Active-standby

BIG-IP APM behavior differs depending on the deployment mode used, the user experience, and with respect to a failover event.

SessionDB employs a “shared-nothing” partitioning scheme to store session entries across all available blades in a VIPRION system. This scheme addresses the challenge of scaling to support high throughput and large data sets.

When storing an entry, SessionDB uses a hashing algorithm on the session entry’s key to determine which available blade it should be stored on. The same hashing algorithm is used to retrieve the entry.

Note: The same key may produce different hash values when the number of available blades in the VIPRION system changes.

Standalone VIPRION

In a standalone multi-blade VIPRION system, where cluster mirroring is set to **Within cluster**, each session ID entry is mirrored to a different blade within the same VIPRION system to mitigate data loss should a blade go offline.

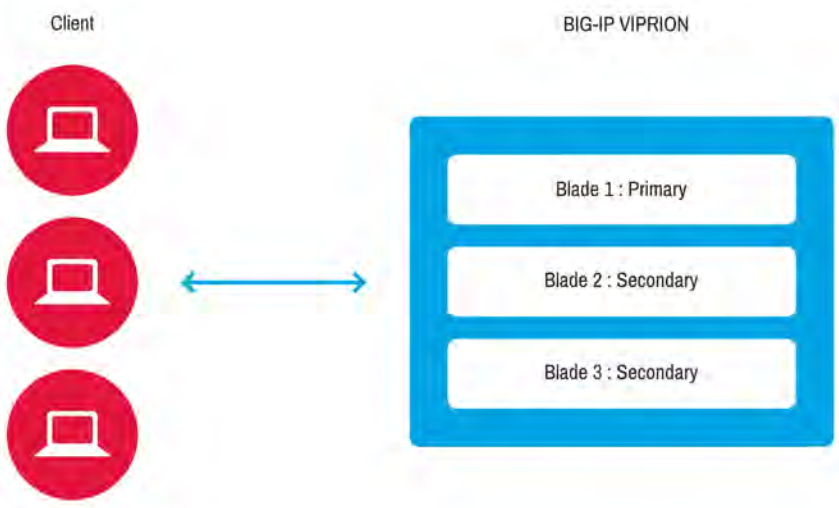

Figure 6.2: Standalone VIPRION cluster with all blades online

In the case of a blade failure, no user sessions are interrupted or lost.

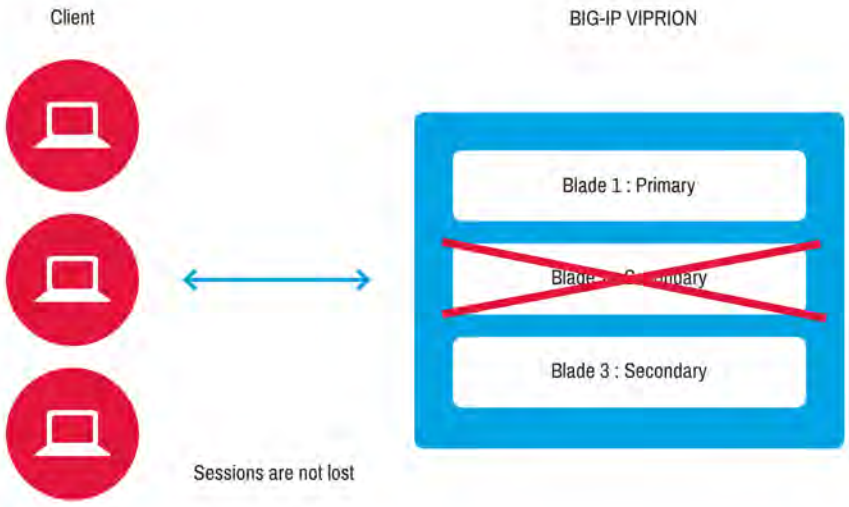

Figure 6.3: Standalone VIPRION cluster with blade 2 offline

VIPRION BIG-IP APM on VIPRION Sync-Failover

In an active/standby sync-failover device group where cluster mirroring is set to **Between cluster**, session entries are distributed among the online blades, then mirrored to the standby VIPRION system.

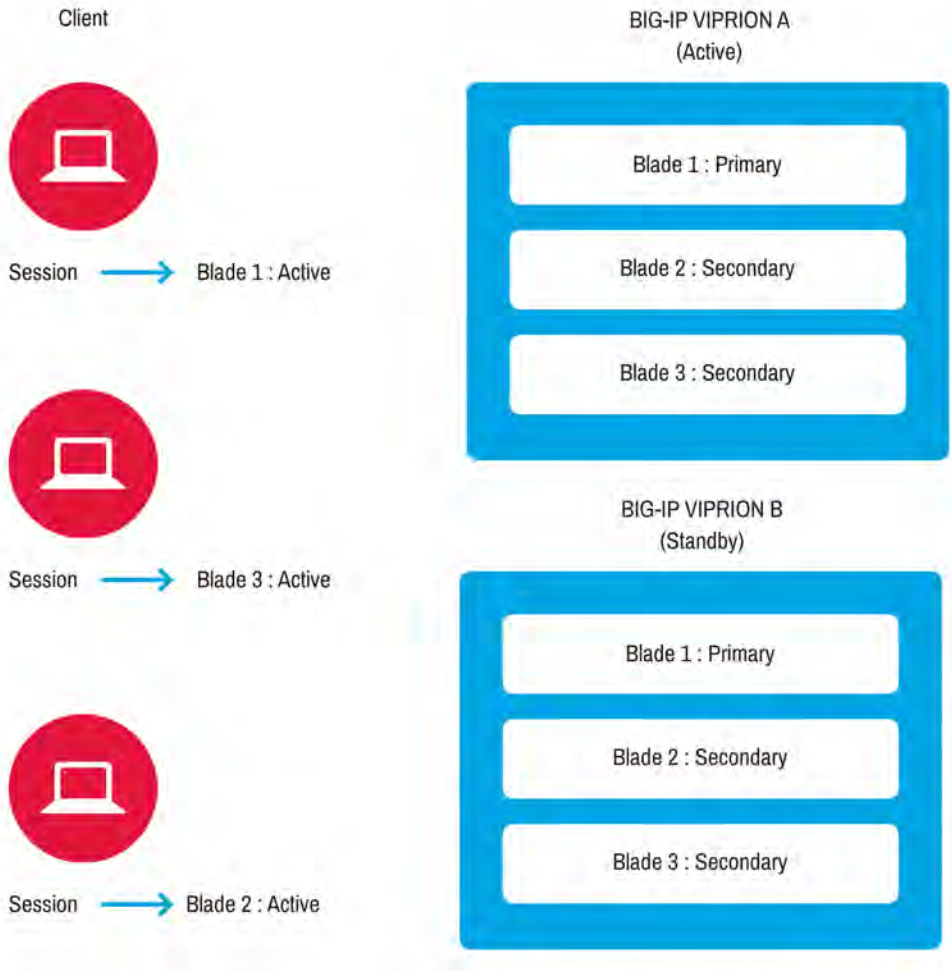

Figure 6.4: Active-standby VIPRION device group with all blades online

If a blade in the active system goes offline, such as BIG-IP VIPRION - A: Blade 2 in the following figure, all the session entries hosted by that blade will be lost. Backup copies of session entries are available on the standby VIPRION's primary blade.

If the active VIPRION system were to maintain its role, existing sessions may need to be re-established when:

- The session entries held in offline blade are lost.
- Existing session entries cannot be found and the session lookup algorithm has changed due to the new blade configuration.

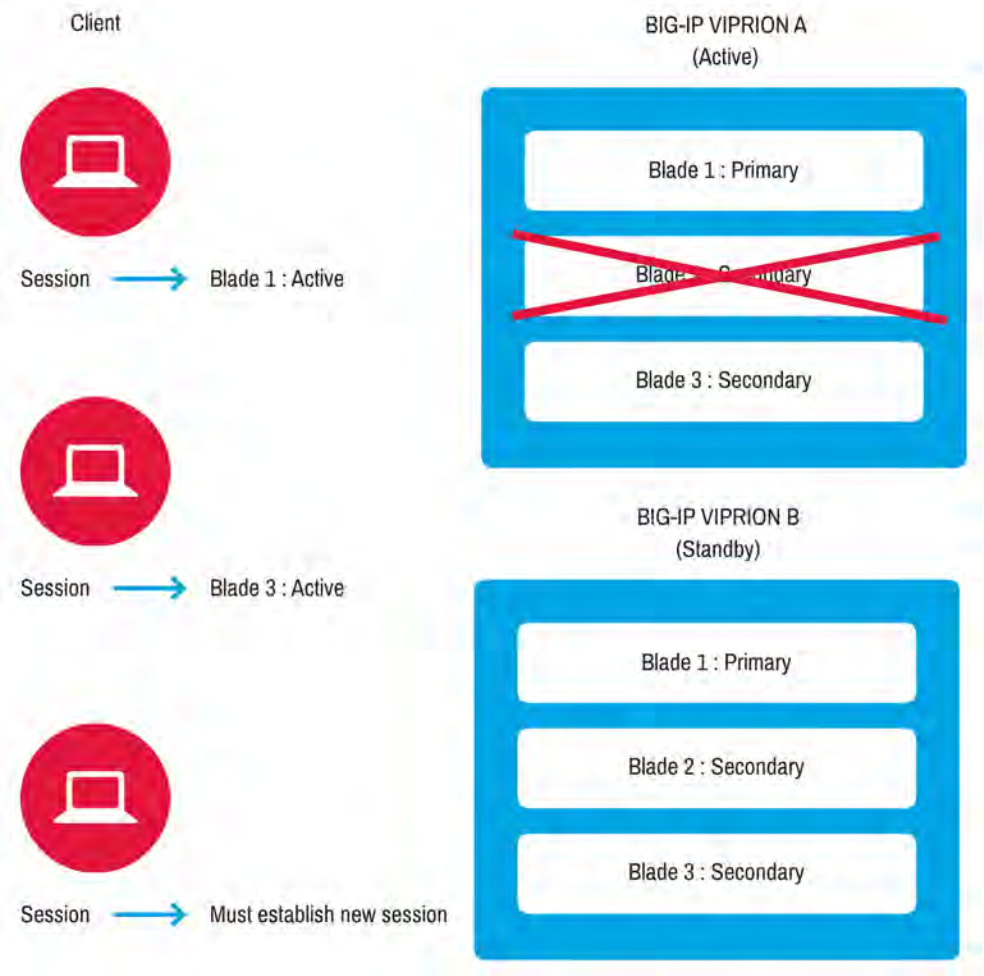

Figure 6.5: Active-standby VIPRION device group with Blade 2 on VIPRION A offline

Note: To minimize user disruption, F5 recommends that a failover be triggered and that the standby VIPRION assume the primary traffic-processing responsibility.

The active/standby sync-failover device group can be configured to trigger the failover automatically. To do this, change the **Minimum Number of Blades Up Before Device Is Considered Available** option to match the number of blades available in the VIPRION chasis.

For example, if there are 2 configured blades in the VIPRION chassis, the value of **Minimum Number of Blades Up Before Device Is Considered Available** should be set to 2.

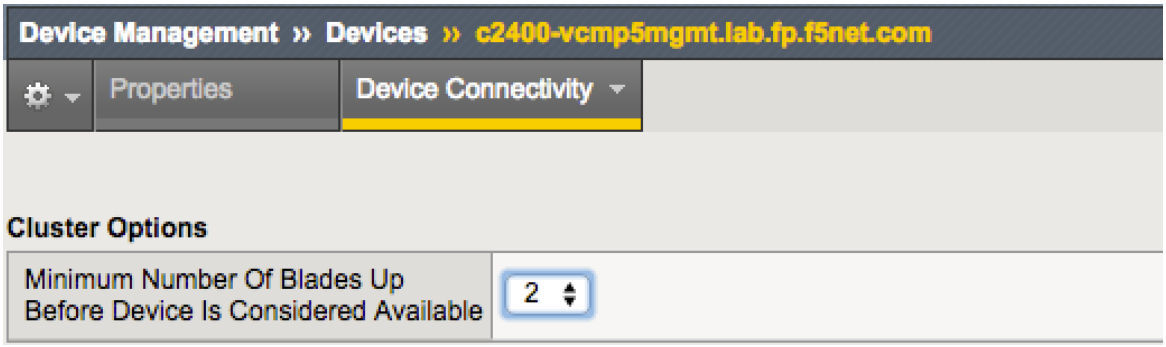

Figure 6.6: Cluster options view of Device Connectivity tab on Device Management > Devices page

User sessions mirrored to online blades remain available by manually forcing a failover to the standby VIPRION system. Existing user sessions are maintained and disruptions minimized.

Management

Introduction

License usage monitoring

Logs

SNMP monitoring

Authentication resource monitoring

Introduction

Several BIG-IP APM management tasks must be consistently performed to maintain the health of the system. These include the following:

- Tracking the number of concurrent user sessions.
- Monitoring the authentication server pool to make sure that valid servers are used to authenticate and authorize users.
- Maintaining and reviewing log files to track usage patterns and other information.
- Preventing disk partitions from filling up, which can degrade performance of the BIG-IP system.

Note F5 recommends remote logging using High Speed Logging with iRules to conserve disk space. Additionally, storing logs externally to the BIG-IP system allows them to be kept for longer periods, which makes long-term trend analysis possible.

License usage monitoring

Monitoring BIG-IP APM resource usage is important to maintaining system health. For every user session with BIG-IP APM, an access license is consumed. If the user is allowed access to a remote access resource, such as network access (VPN), portal access (HTTP tunneling) or application access (AppTunnels), a concurrent user (CCU) license is also consumed.

For more information, refer the *Session Licenses* chapter in this guide.

To make sure that there are always sufficient resources available to the user, F5 recommends periodical monitoring of the available access and concurrent licenses.

For more information, see AskF5 article: ***SOL15032: Determining license limits of the BIG-IP APM system***.

Collecting usage data

BIG-IP APM supports use of SNMP to determine the number of concurrent user licenses in use.

When using SNMP is not possible, you can use the Visual Policy Editor (VPE) to add a policy agent in the access policy to collect license usage data. The figure below shows a **Variable Assign** agent used.

The VPE approach:

- Collects "Concurrent user licenses in use" values only.
- Prevents network resource access.
- Terminates in **Deny.**

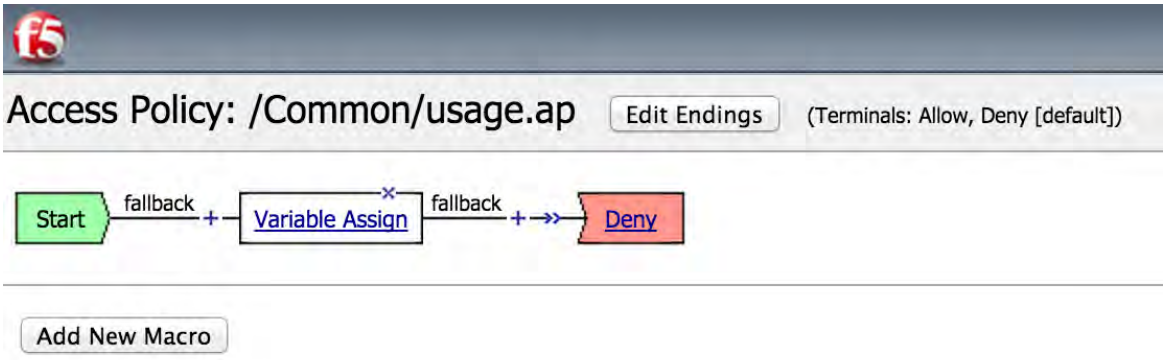

Figure 7.1: Variable Assign access policy agent used to collect license usage

Variable assign agent

The **Variable assign** agent creates four session variables, populated with the TMM license information. The variables include:

- Total Number of Access Session Licenses Available.
- Number of Access Session Licenses Currently in Use.
- Total Number of Concurrent User Licenses Available.
- Number of Concurrent User Licenses Currently in Use.

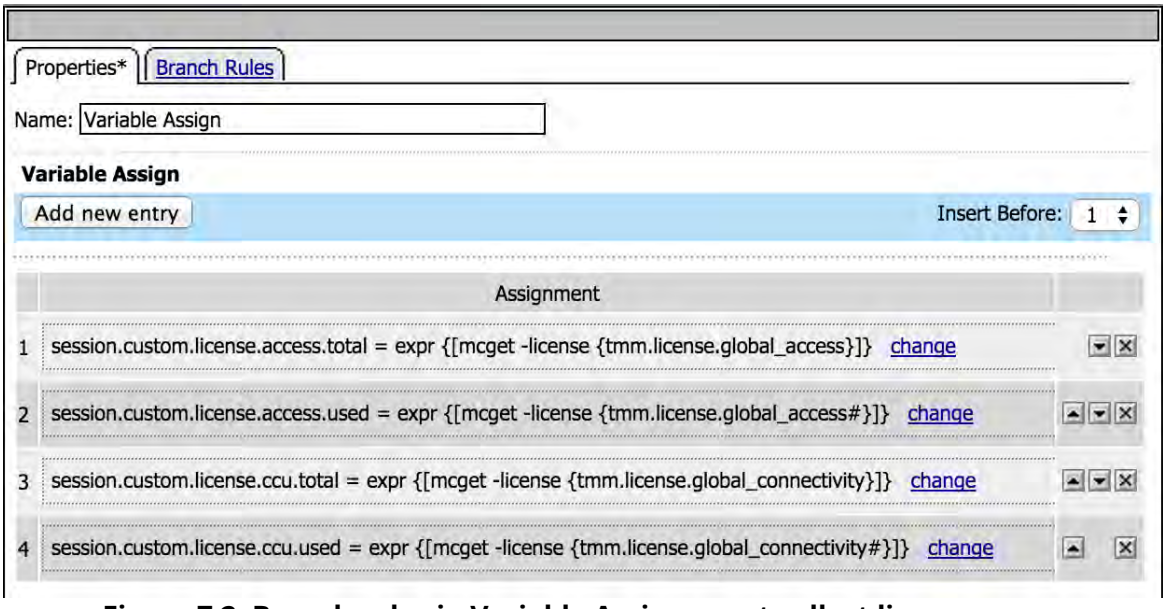

Figure 7.2: Branch rules in Variable Assign agent collect license usage information in session variables

Once these values are stored in session variables, they can be used by the policy and displayed in the VPE. Rather than creating a Message Box action to display these values, they are displayed on the Deny page, using customization shown here.

The following figure shows the **Error message** in the customization field for **Deny**. The variables created in the **Variable Assign** action are used in the error message, but they must use the %{variable-name} syntax so that the value of the variable is displayed, rather than the variable name.

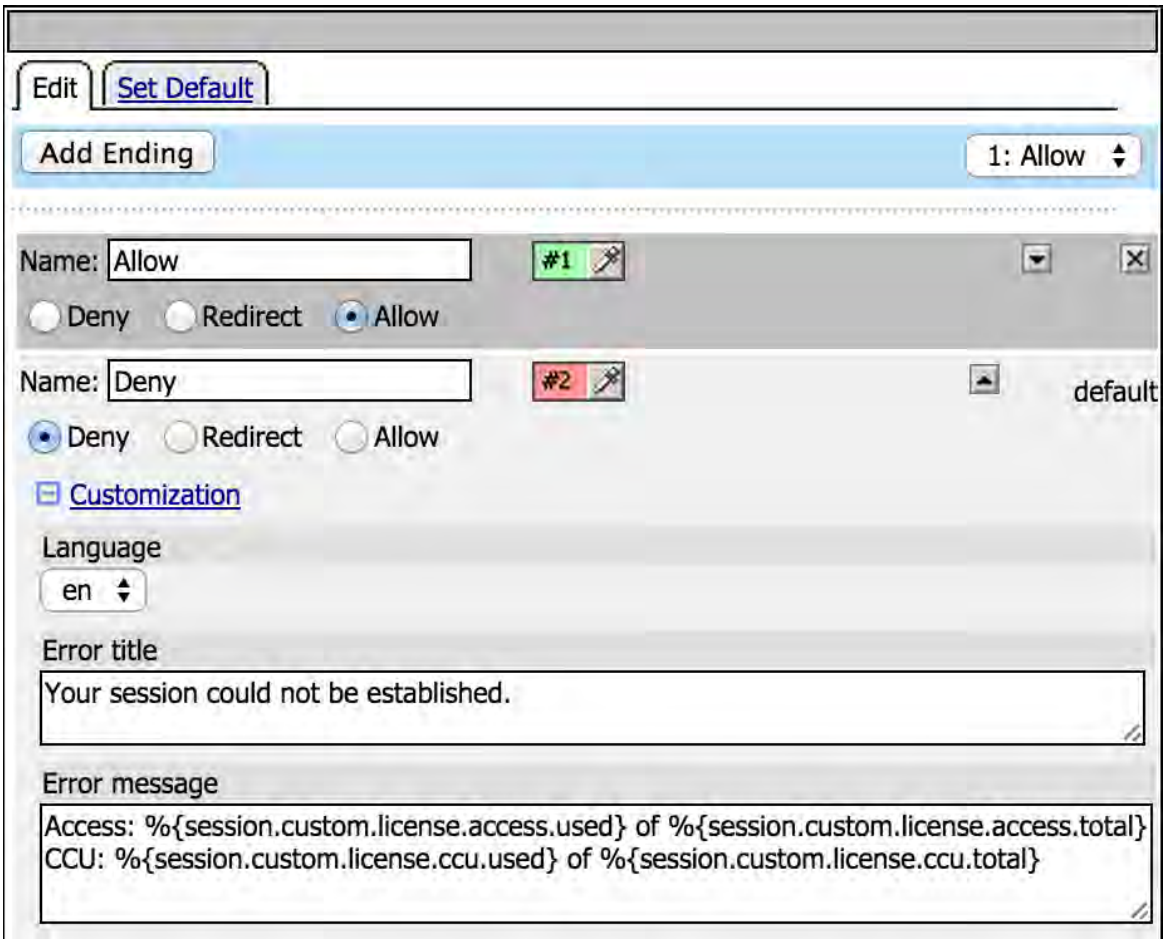

Figure 7.3: Logout page error message configuration

The following figure shows the error, as displayed to a user connection to the web page is denied. It shows one access license out of 2500 in use and zero CCU licenses out of

1500 in use. The one access licence was consumed by session 87e8ee7d. Because this session did not include a remote access resource, no CCUs were consumed.

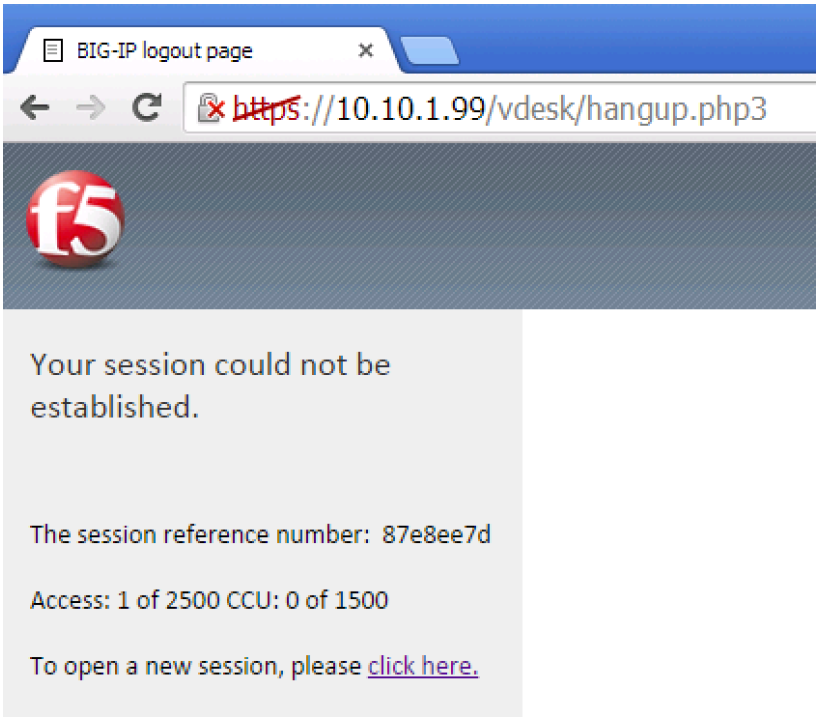

Figure 7.4: Logout page example as seen by user

Tip The license information in the previous example has been made available with a minimal access policy. Development of an automated script is possible but outside the scope of this document.

Logs

Local database

The local database records user session data such as the sessionid, virtual IP address, and client IP address.

To manage the general properties of the local database

1. Go to **Access Policy** > **Reports** > **Preferences**.
2. Configure the preferences:

- **Write to Local Database** (enabled by default) stores log files in the reporting log database on the BIG-IP system. BIG-IP APM reports use of the local database. If you disable this setting, reports are empty or include only data written to the local database before this setting was disabled.
- **Log Rotation Period** indicates the number of days before the local database logs rotate. Logging is available when **Write to Local Database** is enabled. The allowed range is from 0-90 days. If set to 0 (default), logs are rotated based on the number of records configured in the **Maximum Number of Log Entries** option.
- **Maximum Number of Log Entries** indicates the maximum number of local database log records to store. It is available when you enable **Write to Local Database**. The oldest log records are deleted after the specified number of log records is reached. The allowed range is 100,000-5,000,000.
- **Optimize for Reporting** (disabled by default) improves reporting performance through the use of indexes on log data tables. Indexes improve the speed at which records are retrieved from the database at the expense of slowing down the speed at which records are written to the database. In some cases, when speed is prioritized it is preferable to disable indexes. To do this, deselect this option to disable indexes from the log data tables. Changes only take effect after restarting either the logd or the BIG-IP APM system.
- **Log Database Maintenance** clears records from the local database when **Delete** is clicked.
- **Write to APM Log File** (enabled by default) stores logs to **/var/log/apm**.

Because BIG-IP APM reporting utilizes the local database and the Maximum Number of Log Entries allows up to 5,000,000 entries, reporting on a heavily utilized BIG-IP APM system may be very limited.

Important: Local database entries can consume as much as 4 GB of disk space. The file system containing the local database (**/var/lib/mysql**) is limited to 12GB total. Checking available free space using tools such as the df –h command and/or SNMP should be part of a routine maintenance schedule.

High-speed logging

Some administrators may want to log additional data such as when an access session is started or completed. Because excessive logging on the BIG-IP APM system can impair performance, the high-speed logging (HSL) feature can be used to send logs to a remote logging server. HSL utilizes TMM for faster processing and bypasses the local syslog-ng instance altogether. This can yield a performance gain over normal logging by orders of magnitude.

Implementing HSL on the BIG-IP APM system requires the use of iRules. An individual iRule is associated with the virtual server configured with the BIG-IP APM access policy. For example, the following iRule can log when an access session is started, completed, closed, or simply log every HTTP request traversing the access policy:

```
when RULE_INIT {
    ## user-defined: HSL pool
    set static::HSL_POOL "hsl_pool"

    ## user-defined: log ACCESS session start (0 off, 1 on)
    set static::ACCESS_START 1

    ## user-defined: log ACCESS session complete (0 off, 1 on)
    set static::ACCESS_COMPLETE 1

    ## user-defined: log ACCESS session requests (0 off, 1 on)
    set static::ACCESS_REQUEST 1

    ## user-defined: log ACCESS session closed (0 off, 1 on)
    set static::ACCESS_CLOSED 1
```

```
    ## user-defined: username session variable
    set static::ACCESS_USER_VAR "session.logon.last.username"
}
when CLIENT_ACCEPTED {
    set hsl [HSL::open -proto UDP -pool $static::HSL_POOL]
}
when ACCESS_SESSION_STARTED {
    if { $static::ACCESS_START } {
        HSL::send $hsl "<190> ACCESS session started|CLIENT:[IP::client_addr]|VS:
[IP::local_addr]|ID:[ACCESS::session data get session.user.sessionid]"
    }
}
when ACCESS_POLICY_COMPLETED {
    if { $static::ACCESS_COMPLETE } {
        set user ""
        if { [ACCESS::session data get $static::ACCESS_USER_VAR] ne "" } {
            set user "|User:[ACCESS::session data get $static::ACCESS_USER_VAR]"
        }
        HSL::send $hsl "<190> ACCESS session complete|CLIENT:[IP::client_addr]|VS:
[IP::local_addr]${user}|ID:[ACCESS::session data get session.user.sessionid]|RESULT:
[ACCESS::policy result]"
    }
}
when ACCESS_ACL_ALLOWED {
    if { $static::ACCESS_REQUEST } {
        set user ""
        if { [ACCESS::session data get $static::ACCESS_USER_VAR] ne "" } {
            set user "|User:[ACCESS::session data get $static::ACCESS_USER_VAR]"
        }
        HSL::send $hsl "<190> ACCESS session request|CLIENT:[IP::client_addr]|VS:
[IP::local_addr]${user}|ID:[ACCESS::session data get session.user.sessionid]|URI:
[HTTP::uri]"
    }
}
when ACCESS_SESSION_CLOSED {
    if { $static::ACCESS_CLOSED } {
        set user ""
        if { [ACCESS::session data get $static::ACCESS_USER_VAR] ne "" } {
            set user "|User:[ACCESS::session data get $static::ACCESS_USER_VAR]"
        }
        set hsl [HSL::open -proto UDP -pool $static::HSL_POOL]
        HSL::send $hsl "<190> ACCESS session closed${user}|ID:[ACCESS::session data get
session.user.sessionid]"
    }
}
```

This previous iRules example allows the administrator to turn the logging on and off for each individual event by setting the user defined static variables at the top to "1" (on) or "0" (off) and saving.

Caution: This example may not be suitable for your configuration. iRules should be customized for the specific environment and thoroughly tested in a lab environment before placing into production.

SNMP monitoring

Simple Network Management Protocol (SNMP) is an industry-standard protocol that gives a standard SNMP management system the ability to remotely manage and monitor a device on the network. BIG-IP APM supports SNMP v1, SNMP v2c, and SNMP v3.

SNMP can be used to monitor:

- BIG-IP APM sessions
- BIG-IP APM CCU sessions

For more information on how to configure SNMP on BIG-IP, see the ***Configuring SNMP*** chapter in the ***Configuration Guide for BIG-IP Access Policy Manager***

The following table highlights some of the BIG-IP APM SNMP OIDs of interest to monitor:

```
  Task                            OID
  ------------------------------- ------------------------------------------------------
-
   Current Active Access Sessions   F5-BIGIP-APM-MIB::apmAccessStatCurrentActiveSessions
   Current In Progress Sessions     F5-BIGIP-APM-MIB::apmAccessStatCurrentPendingSessions
   Total of Allow Sessions          F5-BIGIP-APM-MIB::apmAccessStatResultAllow
   Total of Denied Sessions         F5-BIGIP-APM-MIB::apmAccessStatResultDeny
   Total CCU sessions               F5-BIGIP-APM-MIB::apmGlobalConnectivityStatTotConns
   Current CCU sessions             F5-BIGIP-APM-MIB::apmGlobalConnectivityStatCurConns
```

All of the above OIDs are counter values with output similar to the following:

```
snmpwalk -v 2c -c public localhost F5-BIGIP-APM-MIB::apmAccessStatCurrentActiveSessions
```

The command output appears similar to the following example:

```
F5-BIGIP-APM-MIB::apmAccessStatCurrentActiveSessions.0 = Counter64: 104
```

In this command output, note the following:

- **<104>** is the total current access sessions on this device

SNMP monitoring for general system health

Because BIG-IP APM has a lot of reporting and other activity that occurs outside of BIG-IP TMOS, F5 recommends monitoring statistics related to general Linux system health, including disk and memory. This monitoring helps long-term trends, including potential problems before they can cause trouble. Monitoring parameters are available the same way as in normal Linux systems. If possible, use the native monitoring software's Linux template.

If native monitoring software is not available, use the following monitoring procedures.

To display disk monitoring information from the command line

- Type the following command:

```
snmptable -v 2c -c public localhost HOST-RESOURCES-MIB::hrStorageTable
```

To display system processes and per-process memory consumption from the command line

- Type the following command:

```
snmptable -v 2c -c public localhost HOST-RESOURCES-MIB::hrSWRunTable
```

```
snmptable -v 2c -c public localhost HOST-RESOURCES-MIB::hrSWRunPerfTable
```

Monitoring these SNMP parameters together can help spot memory leaks in processes, or potential disk space consumption issues.

Authentication resource monitoring

The BIG-IP APM system can be configured to use a single authentication (AAA) server for authentication or a pool of AAA servers for high availability.

Pools can be created for the following AAA resource types:

- RADIUS
- Active Directory
- LDAP
- CRLDP
- TACACS+

A BIG-IP monitor should be assigned to the AAA pool in order to determine which servers are available to receive authentication requests. BIG-IP APM does not load-balance authentication requests between the pool members, but instead by the priority number assigned to the pool member. Authentication requests are serviced by the next highest priority pool member if the currently active server is unavailable.

Tip Currently, only the **gateway_icmp** monitor is appropriate for monitoring AAA servers. Select this monitor from **Server Pool Monitor** when creating the AAA Server object.

For more information on priority group activation and other pool related topics, see *About Pools* in ***BIG-IP Local Traffic Management: Basics***.

Access programmability

Introduction

Access iRules structure

Visual Policy Editor programming

Clientless mode

Introduction

iRules is a powerful and flexible feature of BIG-IP devices based on F5 TMOS architecture. iRules provides you with unprecedented control to directly manipulate and manage any IP application traffic. Using an easy to learn scripting syntax, iRules can perform nearly any traffic function for network traffic passing through a BIG-IP system, including routing, re-routing, redirecting, inspecting, modifying, delaying, discarding, rejecting, and logging.

iRules and F5 support

F5 provides basic support for existing iRules. Support can assist with checking iRules syntax, troubleshooting specific commands of iRules functionality, and validating iRules logic. iRules must have been previously operating prior to contacting F5 support. F5 support will not provide concept, design, authoring, or creation of iRules.

DevCentral community

DevCentral is an online developer community of more than 160,000+ F5 users worldwide who collaborate and share innovations, including code samples, new techniques, and other tips. DevCentral is also the home of the *iRules Wiki*, the location of iRules reference documentation and a great place to visit when you are getting started with iRules.

iRules on demand and F5 Professional Services

iRules On-Demand supplies custom-developed iRules to address the specific and unique needs of each customer. Visit the ***F5 Professional Services*** page (f5.com/support/professional-services) to review offerings.

ACCESS iRules structure

BIG-IP APM iRules can be broken down into two primary components: **access events** and **access commands**.

Access events

iRules events are programming structures that are triggered within the context of a certain state of a connection. With respect to a BIG-IP APM access session, events are triggered at different stages of the access policy initiation, evaluation, completion, and termination.

For example, when a user initiates a new access session, an event is triggered. When a user completes access policy evaluation, an event is triggered. Throughout policy evaluation and upon subsequent "allowed" access requests, events are triggered. In this way BIG-IP APM iRules provide a robust mechanism for programmatically controlling nearly every aspect of the authentication and access process.

The figure below shows access events in the context of access policy evaluation.

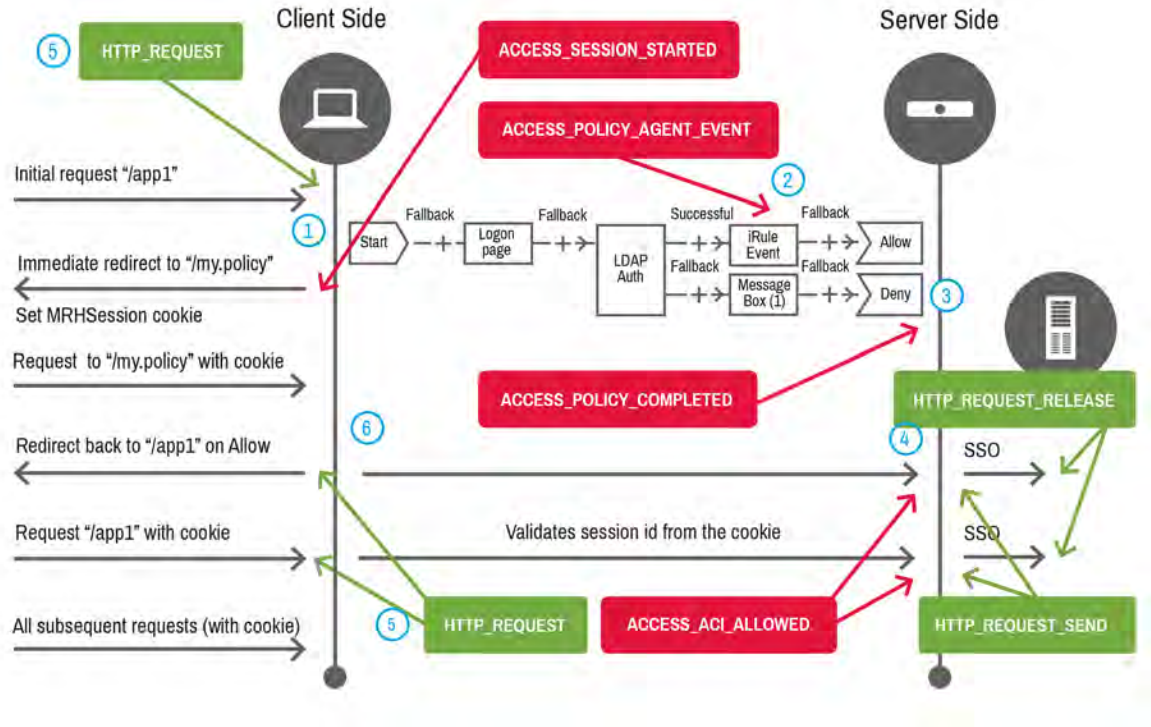

Figure 8.1: Access iRules event diagram

1. The client browser or the BIG-IP Edge Client application makes an initial request to a BIG-IP APM virtual server.
In that request, the client has no established session with BIG-IP APM and sends no session cookie.
BIG-IP APM creates an access session and immediately redirect the client to a special **"/my.policy"** URI and sets the cookie, **MRHSession**, (pointer) for that access session in the redirect response.
The **ACCESS_SESSION_STARTED** event is triggered. Information about the client and session are available here, including IP addresses, browser and client type, and session ID.

2. If an iRule agent is found at any point in the access policy evaluation, the **ACCESS_POLICY_AGENT_EVENT** event is triggered.
The **ACCESS_POLICY_AGENT_EVENT** allows access policy processing to move into

an iRules event that has full access to all of the session data collected up to that point.
Multiple iRules event agents can exist in an access policy.

3. At the end of access policy evaluation, the **ACCESS_POLICY_COMPLETED** event is triggered.
The **ACCESS_POLICY_COMPLETED** event has all of the information collected from the evaluation, as well as the result of policy evaluation (**Allow** or **Deny**).

4. After policy evaluation, and during all following requests, the **ACCESS_ACL_ALLOWED** event is triggered.
This **ACCESS_ACL_ALLOWED** has access to all of the session information collected from the evaluation, as well as HTTP request information. This event can be thought of as an **HTTP_REQUEST** event triggered after a completed access policy.

5. **HTTP_REQUEST** events are still triggered and may contain access session information. Take care to identify the **HTTP_REQUEST** before attempting to manipulate access session information.
For example, an **HTTP_REQUEST** event is triggered before the **ACCESS_SESSION_STARTED** event and before any **ACCESS_ACL_ALLOWED** event. The first **HTTP_REQUEST** event does not contain information about the evaluated policy and user while following **HTTP_REQUEST** events may contain that information. It is easier and safer to use the access events directly. F5 recommends that process rather than use of **HTTP_REQUEST** events if possible.

6. During access policy evaluation, access to **/my.policy** URI is allowed. If access policy agents such as **Logon Page** and **Message Box** interact with the client , the **HTTP_REQUEST** event is triggered but it is hidden by default. To unhide these events, use the **ACCESS::restrict_irule_events** command.

Important Attempting to manipulate the access session in intermediate **HTTP_REQUEST** events can cause unexpected behavior.

For more information, see the ***F5 DevCentral Access wiki***.

Access commands

The access commands allow for direct manipulation of session and policy information during and after policy evaluation. iRules such as the following example can be used to retrieve logon user information from the session and send that as an HTTP header to the backend application on each HTTP request after policy evaluation.

```
when ACCESS_ACL_ALLOWED {
     set user [ACCESS::session data get "session.logon.last.username"]
     HTTP::header insert "X-USERNAME" $user
}
```

For more information, see the ***F5 DevCentral Access wiki***.

Session variables

When an access session starts, data about the session begins to be collected in a discrete and secure message cache. As the policy evaluates, more information is stored in that cache. During access policy evaluation, if a decision has to be made, those collected session variables are examined.

Session variables have a hierarchical naming convention. For example:

```
session.logon.last.username
```

There is no restriction on the name itself and it isn't a member of an enumerable array. You can for example create a session variable named "bob", but without the "session" prefix it would not show up in session reports.

F5 recommends naming session variables in context to what they represent. The **session.logon.last.username** variable represents a username value, collected at logon by an agent like a logon page.

For more informatin, see *Session Variables* in ***BIG-IP Access Policy Manager: Visual Policy Editor***.

Every policy agent is responsible for either creating session variables or evaluating them. You don't have to know anything about session variables to create or implement a

typical access policy. However, the ability to access and manipulate session variables can be very useful when troubleshooting.

View access session variables

There are a few different ways you can view created session variables:

- View access policy reports
- Use **sessiondump**
- View logs
- View message boxes
- Use iRules

View session variables in access policy reports

F5 recommends viewing session variables using this method.

To view session variables in access policy reports using the Configuration utility

1. Go to **Access Policy** > **Reports** > **View Reports**.
2. Under **Report Parameters**, set a time period and click **Run Report**.
3. On the **All Sessions** tab, click the **View Session Variables** link for the report you want to view.

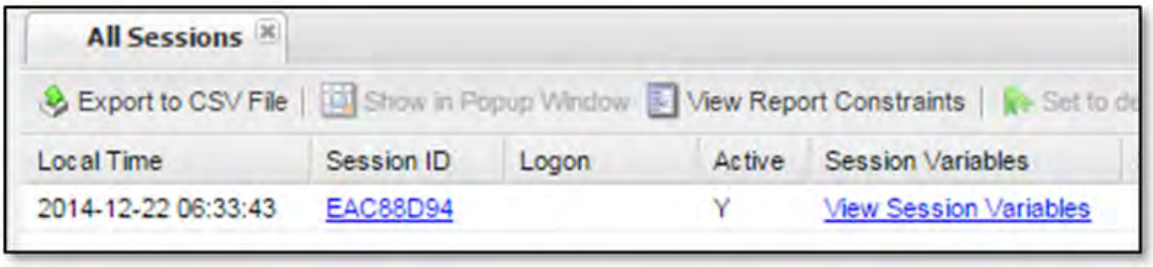

Figure 8.2: All Sessions tab in session reports interface

Session variables for your report open in a new tab. The variables are presented in a hierarchical display.

Figure 8.3: Session Variables report tab

View session variables with sessiondump

The **sessiondump** utility is a command line alternative you can use to view logs. The utility has a several available commands, including the following:

Table 8.1 sessiondump commands

Command	Result
-list	Presents a short list of active sections, one session per line.
-allkeys	Presents all session variables for all active sessions. Use sparingly on busy systems.
<Session ID>	Presents session variables for the session ID you enter. The session ID is an 8-character string.

The following figure shows the return syntax of the **-allkeys sessiondump** utility command. It may be large if used on a busy system. However, it can be scripted. For

example, to display a list of all of the authenticated users, you could filter on the **session.logon.last.username** session variable using the **grep** command:

```
sessiondump -allkeys | grep session.logon.last.username
```

```
[root@bigip-internal:Active:Standalone] config # sessiondump -allkeys
209b679c 10 SessionKey
209b679c.session.access.profile 27 /Common/simple-logon-policy
209b679c.session.access.profiletype 1 0
209b679c.session.assigned.uuid 41 tmm.uuid./Common/simple-logon-policy.fred
209b679c.session.client.activex 1 0
209b679c.session.client.browscap_info 87 uimode=0&ctype=Mozilla&cversion=5&cjs=1&cactivex=0&cplugin=1&cplatform=Win7&cpu=unknown
209b679c.session.client.cpu 7 unknown
209b679c.session.client.js 1 1
209b679c.session.client.platform 4 Win7
209b679c.session.client.plugin 1 1
209b679c.session.client.type 7 Mozilla
209b679c.session.client.version 1 5
209b679c.session.createdfrom 6 ACCESS
209b679c.session.end 9 timed_out
209b679c.session.ha_unit 32 f883a0e4e78d00ec2470ef447b4da00f
209b679c.session.inactivity_timeout 3 900
209b679c.session.logon./Common/simple-logon-policy_act_logon_page_ag.logonname 4 fred
209b679c.session.logon./Common/simple-logon-policy_act_logon_page_ag.result 1 1
209b679c.session.logon./Common/simple-logon-policy_act_logon_page_ag.username 4 fred
209b679c.session.logon.last.logonname 4 fred
209b679c.session.logon.last.result 1 1
209b679c.session.logon.last.username 4 fred
209b679c.session.logon.page.customization.group 45 /Common/simple-logon-policy_act_logon_page_ag
209b679c.session.logon.page.errorcode 1 0
209b679c.session.policy.result 5 allow
209b679c.session.rest.clearcache 1 0
209b679c.session.rest.groupname 0
209b679c.session.rest.requestdomain 0
209b679c.session.rest.requesttype 0
209b679c.session.rest.username 0
209b679c.session.server.landinguri 1 /
209b679c.session.server.listener.name 26 /Common/test.domain.com-vs
209b679c.session.server.network.name 15 test.domain.com
209b679c.session.server.network.port 2 80
209b679c.session.server.network.protocol 4 http
209b679c.session.snapshotid 32 71c39630815_loooooooooooooooo
209b679c.session.state 5 allow
209b679c.session.stats.bytes.in 1 0
209b679c.session.stats.bytes.out 1 0
209b679c.session.stats.egress.compressed 1 0
209b679c.session.stats.egress.raw 1 0
209b679c.session.stats.ingress.compressed 1 0
209b679c.session.stats.ingress.raw 1 0
209b679c.session.stats.packets.in 1 0
209b679c.session.stats.packets.out 1 0
209b679c.session.ui.lang 2 en
209b679c.session.ui.mode 1 0
209b679c.session.user.agent 90 Mozilla/5.0 (Windows; U; Windows NT 6.1; en-US; rv:1.9.2.13) Gecko/20101203 Firefox/3.6.13
```

Figure 8.4: Session variables displayed using -allkeys command in sessiondump

View session variables in logs

The Visual Policy Editor (VPE) contains a **Logging** agent that will allow you to define a subset of session variables to capture in the BIG-IP APM log. You can add this agent into your access policy and then define it using the variable you're looking for.

The **Logging** agent allows you to define a message in the **Log Message** field and a wildcard match on based on a hierarchy of variables. In the following example, a custom

log is configured to capture **session. logon.last.*** variables created by the **Logon Page** policy agent.

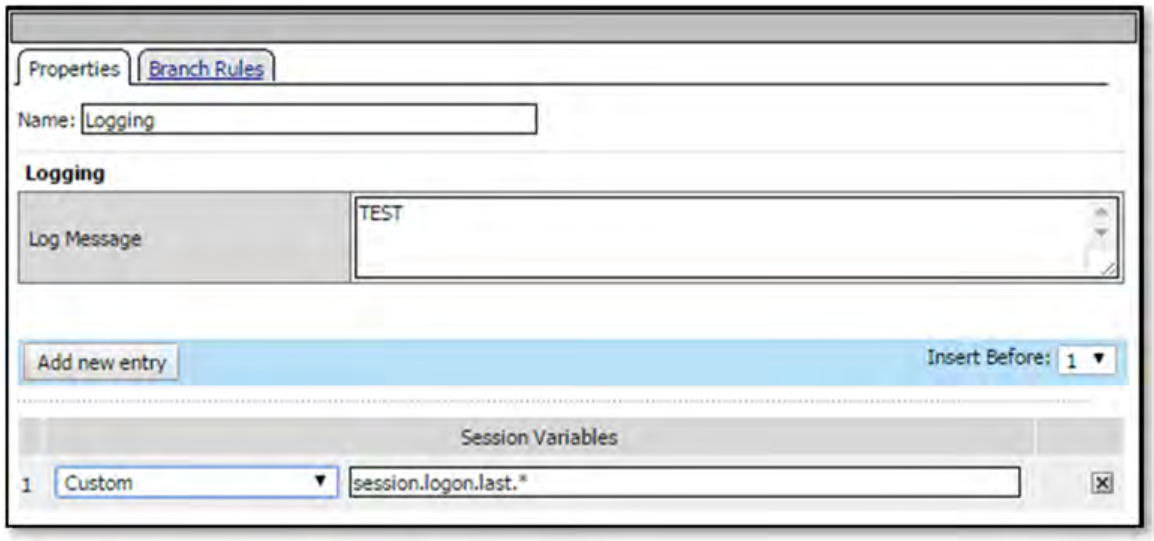

Figure 8.5: Access policy Logging agent Properties tab configuration

To view the log file from the command line

- Type the following command:

```
tail -f /var/log/apm
```

The contents of the BIG-IP APM log files will stream. Log files may be large. You can filter the results using the grep command. For example, if you want to find the session.logon.last.logonname variable, you can to use the grep command to look for the variable that follows log message **TEST**. In this example, you would type the following syntax:

```
tail -f /var/log/apm |grep -A4 TEST
```

The **TEST** string occurs one line immediately preceding the **session.logon.last.logonname** so the **-A4** portion of the input will return the TEST string as well as the 4 lines that follow it.

```
Dec 22 06:57:23 ltmsec notice apd[1457]: 01490143:5: 2f1a92a5: Logging Agent: TEST
Dec 22 06:57:23 ltmsec notice apd[1457]: 01490113:5: 2f1a92a5: session.logon.last.logonname is fred
Dec 22 06:57:23 ltmsec notice apd[1457]: 01490113:5: 2f1a92a5: session.logon.last.password is **********
Dec 22 06:57:23 ltmsec notice apd[1457]: 01490113:5: 2f1a92a5: session.logon.last.result is 1
Dec 22 06:57:23 ltmsec notice apd[1457]: 01490113:5: 2f1a92a5: session.logon.last.username is fred
```

Figure 8.6: Session variable information in BIG-IP APM log messages

This method is often used to target specific information. You could, for example, capture data about user sessions and send that to a remote syslog-capable service like Splunk.

Tip BIG-IP APM log messages are located in the **/var/log/apm** file.

View session variables with message boxes

Using **Message Box** policy agents to view session variables is often deployed during troubleshooting. It is particularly useful if an access policy is functioning other than expected and you cannot diagnose the source of the problem. You can insert one or more **Message Box** agents in the policy path to test the policy. If you want to see a specific value, or set of values, at given points, you can use the following syntax inside the **Message Box** agent:

```
%{session.variable.name}
```

Figure 8.7: Access policy Message Box agent Properties tab configuration

When the policy evaluation occurs, each **Message Box** will be triggered at its place in the policy path and display the defined session variable. The following figure shows a sample of a message returned by the **Message Box** agent configured in the previous figure.

Figure 8.8: Message Box as seen by user

View session variables using iRules

iRules can read and write access session variables using the **ACCESS::session data** structure. The following syntax is an example of reading a session variable in a line of iRules code:

```
set user [ACCESS::session data get session.custom.user]
```

Create session variables

You can create session variables using the **Variable Assignment** agent in the VPE or using iRules.

Create session variables with Variable Assignment agent

The **Variable Assignment** agent supports creation of custom variables within its interface. In the following figure, the custom variable **session.custom.user** is defined with a text string **bob**.

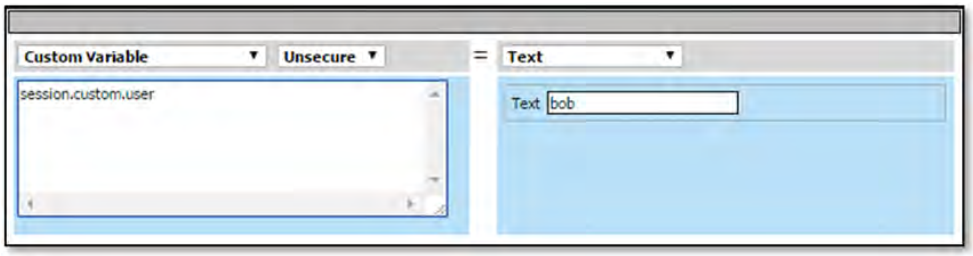

Figure 8.9: Access policy Variable Assignment agent Custom Variable configuration

An access session variable must be hierarchical, but the format is arbitrary. For example, a session variable named **bob** is not allowed but **test.bob** is. However, the first value must be **session** for it to show up in reports view. Therefore, **session.bob** or **session.test.bob** are allowed session variables and will also show up in the reports view. F5 recommends using the prefix **session.custom** when defining custom variables.

Custom expressions are also supported in the **Variable Assignment** agent. These include AAA attributes, other session variables, and custom expressions, as shown in the following figure.

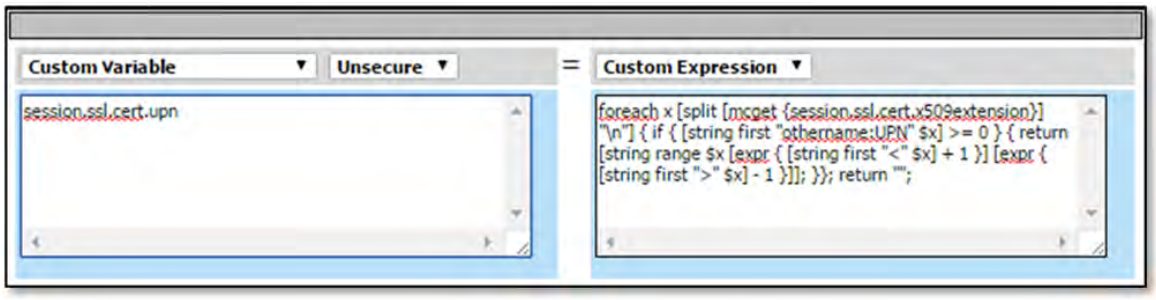

Figure 8.10 Access policy Variable Assignment agent Custom Expression configuration

Create session variables with iRules

The **ACCESS::session data** structure can also write session variables. The following syntax is an example of setting an access session variable (**bob**) in a line of iRule code:

```
ACCESS::session data set session.custom.user "bob"
```

Passwords

Password session variables display as masked if they exist. The **Variable Assignment** agent and iRules **ACCESS::session** construct have the ability to store values encrypted in the session database.

The **Logon Page** agent automatically sets a field of type password as an encrypted session variable. To manually create a secure encrypted session variable using the **Variable Assignment** agent, select **Secure**.

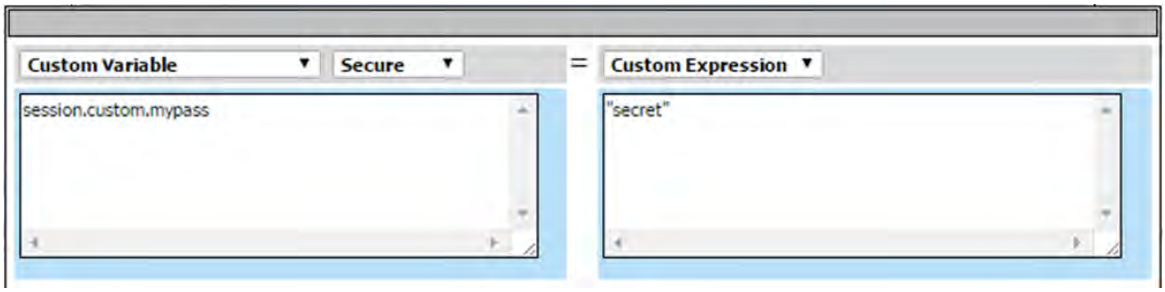

Figure 8.11: Access policy Logon Page agent Secure Custom Variable configuration

To do the same with iRules, use the **-secure** flag in the **ACCESS::session** command, as shown in the following syntax:

```
ACCESS::session data set -secure session.custom.mypass "secret"
```

The **SSO Credential Mapping** agent in the VPE is responsible for decrypting a value from an encrypted session variable. It takes whatever encrypted session variable you're using for a password and sends a decrypted copy of that value to the **session.sso.token.last.password** session variable.

Many of the BIG-IP APM SSO profiles use this new session variable for server-side authentication. You can do this using the **-secure** flag, as shown in the following example:

```
session.custom.mypass = return [mcget -secure {session.logon.last.password}]
```

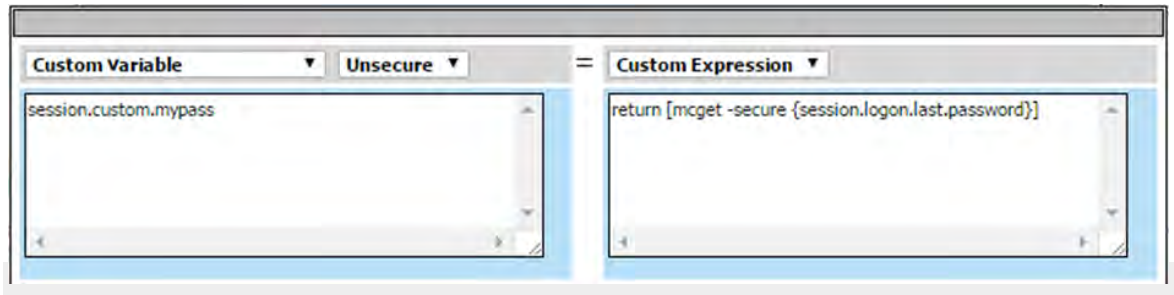

Figure 8.12 : Access policy SSO Credential Mapping agent Unsecure Custom Variable configuration

Once the previous command is run, the message cache will contain a decrypted copy of the password. However, the **session.sso.token.last.password** variable will still display masked in reports.

For more information on access session variables, see ***F5 DevCentral Access wiki***.

Visual Policy Editor

There are two ways to change policy behavior during access policy evaluation: using iRules or using VPE branch rules.

iRules

In order to use iRules in access policy evaluation, an **iRule Event** agent has to be inserted.

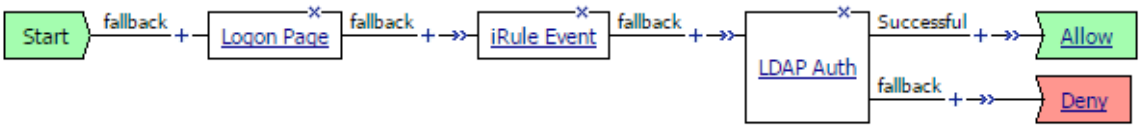

Figure 8.13: Access policy with iRule event agent

The insertion of the **iRule Event** agent triggers an **ACCESS_POLICY_AGENT_EVENT** event in the iRule.

Multiple **iRule Event** agents can be targeted individually by evaluating the access policy **agent_id** access policy value and assigning a unique ID in each iRule event agent.

For example, in the following figure two **iRule Event** agents exist in the policy, each configured with a unique **ID** defined under **Custom iRule Event Agent.**

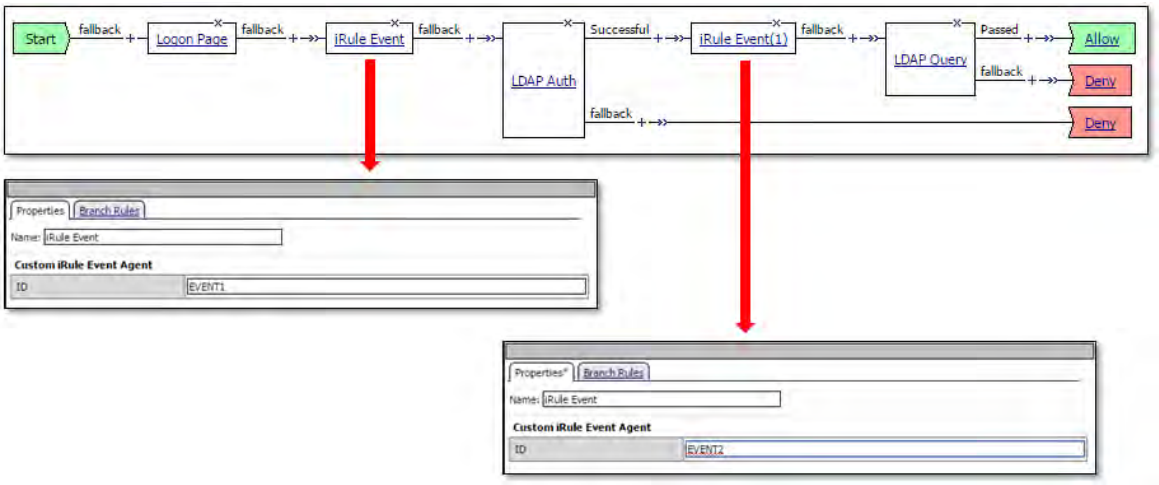

Figure 8.14: Access policy iRule event agent configuration

In the iRule code, perform an evaluation using the following syntax:

```
when ACCESS_POLICY_AGENT_EVENT {
    switch [ACCESS::policy agent_id] {
        "EVENT1" {
            # do something here
        }
        "EVENT2" {
            # do something else here
        }
    }
}
```

In this example, the first iRule event has access to information collected from session initiation and the logon page agent. This information can be used to create and/or stage information for the upcoming LDAP authentication. The second iRule event now has information collected from the **LDAP Auth** agent. This information can be used to create and/or stage information for the upcoming LDAP Query.

Branch rules

Access policy branch rules are written in TCL command language and can be executed in the branch options of any VPE agent. Branch rules are not the same as iRules and do not contain iRules protocol or other iRules-specific commands.

In any of the examples described in the iRules section, agents having multiple output paths use using branch rules to determine the correct path.

For example, the **LDAP authentication** agent shown in the previous figure uses the following branch rule syntax by default to decide if it should follow the **Successful** or **fallback** path:

```
expr { [mcget {session.ldap.last.authresult}] == 1 }
```

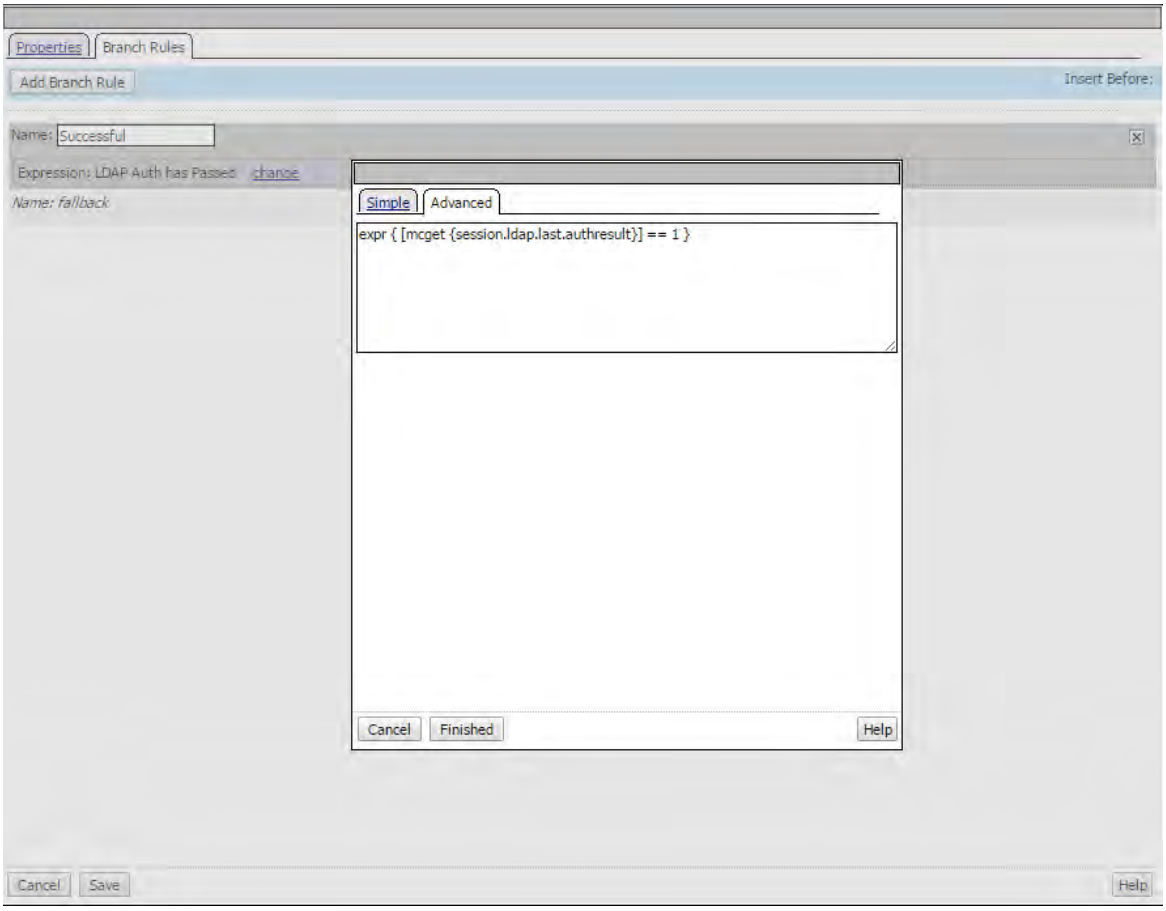

Figure 8.15: LDAP authentication branch rules configuration

In addition, branch rules are also often used to create custom expressions. In the following figure, a branch rule is used as a custom expression.

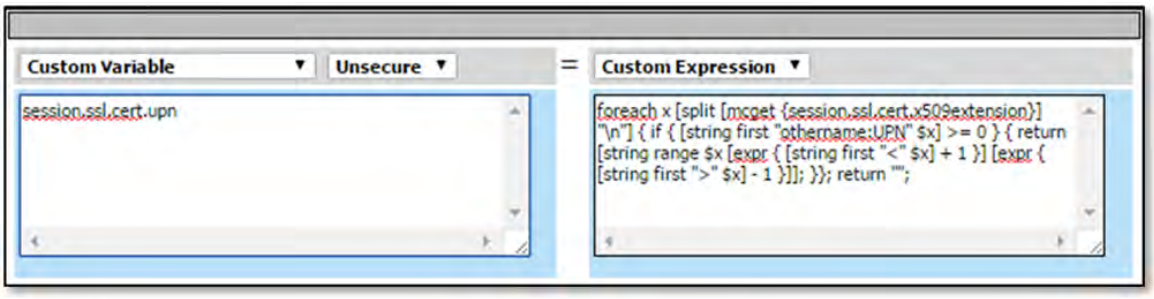

Figure 8.16: Variable assign agent custom expression configuration

Note The custom code field is a simple text block. It optionally supports line breaks and spacing, although these are not recommended. Each line of code must end with a semicolon.

The branch rule in the previous figure uses a custom expression to extract the **userPrincipalName (UPN)** from a client certificate. The following is the branch rule syntax:

```
 foreach x [split [mcget {session.ssl.cert.x509extension}] "\n"] {
      if { [string first "othername:UPN" $x] >= 0 } {
           return [string range $x [expr { [string first "<" $x] + 1}] [expr {string
first ">" -1}]];
      }
 };
 return "";
```

Branch rule syntax

A branch rule includes several elements, including the mcget statement, the **x509** extension, and the **expr** and **return** functions.

Message cache get (mcget) statement

The following is the syntax of the **mcget** statement in the previous example:

```
 [mcget {session.ssl.cert.x509extension}]
```

mcget allows access to session variables from inside branch rules. In this example, it will return the data inside the **session.ssl.cert.x509extension** variable. The data in the session is populated when BIG-IP APM receives a client certificate.

x509 extensions

The **x509 extensions** are a long list of attributes separated by newline characters.

You can break the list using the Tcl **[split]** command and then run through the list with a **foreach** loop.

In each line of the **x509** extensions, if the line contains **othername:UPN**, use a set of string commands to extract this value and return it. That returned value will be assigned to the arbitrary **session.ssl.cert.upn** session variable. It will also be defined on the left side of the **Variable Assignment** agent. If **othername:UPN** is not found, the code will return **""**.

expr and return

Each branch rule will contain one or more **expression** (**expr)** or **return** commands. The **expr** command functions as a Boolean operator. When it returns "true" for an input value, the policy follows the branch. When it returns "false," the next available policy branch is followed. This continues until the policy reaches the **fallback** branch. If more than one branch returns "true," the first "true" policy path executes.

The **expr** command is the same one used in Tcl math, so the following expression works:

```
expr { 10/5 == 2 }
```

expr can also be used to output a value for variable assignment. For example:

```
session.custom.count = expr { [mcget {session.custom.count}] + 1 }
```

The **return** operator returns a values from the message cache using the **mcget** command:

```
return [mcget {session.logon.last.username}]
```

Empty agent

Access policy branches typically contain a branch rules tab. On this tab, existing built-in branch rules or custom branch rules can add functionality to the policy.

If branches need to be built outside an authentication or assigment, the **Empty** agent can be used. The **Empty** agent has no properties and no branches, and it can be configured to trigger and evaluate any policy condition for any reason.

The following figure shows a sample configuration for an **Empty** agent. In this example the agent is called **DG Condition**.

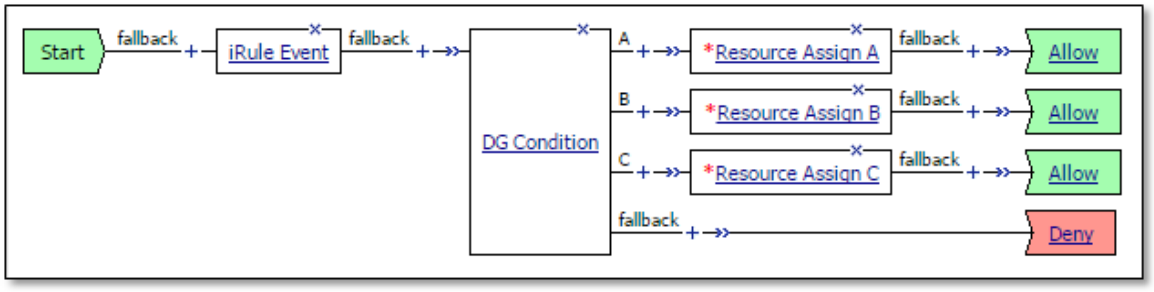

Figure 8.17: Configuring Empty agent

The **DG Condition** box in this example contains three branch conditions to evaluate. The following figure shows the agent's configured branch rules. Each branch evaluates the **session.custom.dgvalue** session variable to test whether the condition is true or false, depending on the input.

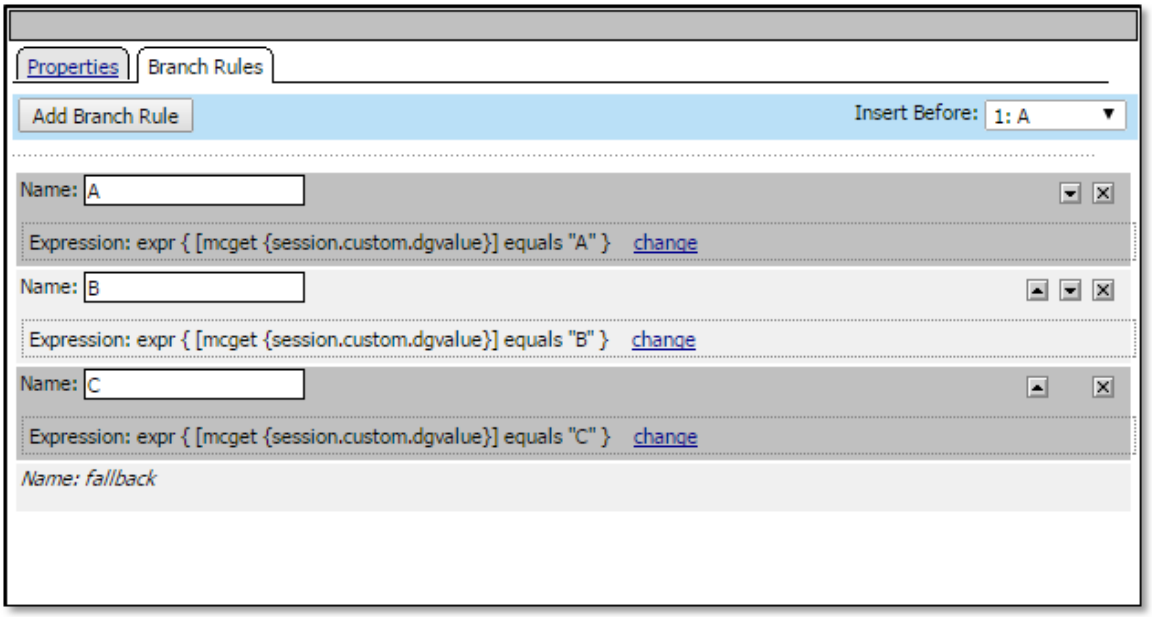

Figure 8.18: Branch rules tab in Empty agent

The branches are processed in order, from top to bottom. If **session.custom.dgvalue** doesn't match the first branch, it processes the next available branch. If none return

"true," it processes the fallback branch. If multiple conditions are "true," the first branch evaluated as "true" will be followed.

For more information on branch rules, see: *Tcl Usage* in ***BIG-IP Access Policy Manager: Visual Policy Editor***.

Branch rules vs. iRules

Branch rules and iRules each have benefits and drawbacks. Circumstance and personal preference wil determine whether you use a branch rules or iRules to manipulate access policy functionality.

Branch rules have the advantage of being part of the access policy. If an access policy is exported, the branch rules are automatically included, while iRules must be exported separately.

iRules can be employed in place of a branch expression, except for agent branch path evaluation. In most cases the iRules will be simpler and cleaner to use than branch rules, as shown in the following syntax.

Taking the branch rule expression from above,

```
 foreach x [split [mcget {session.ssl.cert.x509extension}] "\n"] {
      if { [string first "othername:UPN" $x] >= 0 } {
           return [string range $x [expr { [string first "<" $x] + 1}] [expr {string
first ">" -1}]];
      }
 };
 return "";
```

the same functionality can be performed with iRule as below:

```
 if { [ACCESS::session data get session.ssl.cert.x509extension] contains
"othername:UPN<" } {
      set upn [findstr [ACCESS::session data get session.ssl.cert.x509extension]
"othername:UPN<" 14 ">"]
 }
```

The iRule skips the **foreach** loop and directly extracts the string contents with an iRule **findstr** command.

Clientless mode

"Clientless" refers to an end point which doesn't use a standard web browser style client. This client may not support HTTP cookies, may not follow HTTP redirects or may not have a way to collect user input. These clients may include Citrix receivers, ActiveSync clients, Outlook Anywhere, or other customised web apps or web services.

In a normal web access management/LTM-APM session, the first request is met with a redirect to **/my.policy** and a **Set-Cookie** with the initial **MRHSession** token. Subsequent requests flow between the client and BIG-IP APM while the access policy is being evaluated. At the end of that evaluation, if the session is allowed, a final redirect sends the client back to the originally requested URI.

Clientless mode disables these redirects and adds the **MRHSession** token in the first response from the application instead of the initial redirect to **/my.policy**. Any subsequent requests from the client will use the same session if the client sends back the session cookie. If the client does not send back the session cookie, then a new BIG-IP APM session is created for each request. The existing sessions will be removed based on the timeout settings in the access policy.

The access policy **Logon Page** agent extracts inputs from the request headers or body. No additional input is collected from the user during access policy evaluation. Therefore, you cannot use any agents which interrupt the flow of traffic. Access policy agents requiring user input are ignored. These include **Logon Page**, **End Point Security**, **Message Box**, **Decision Box**, and others.

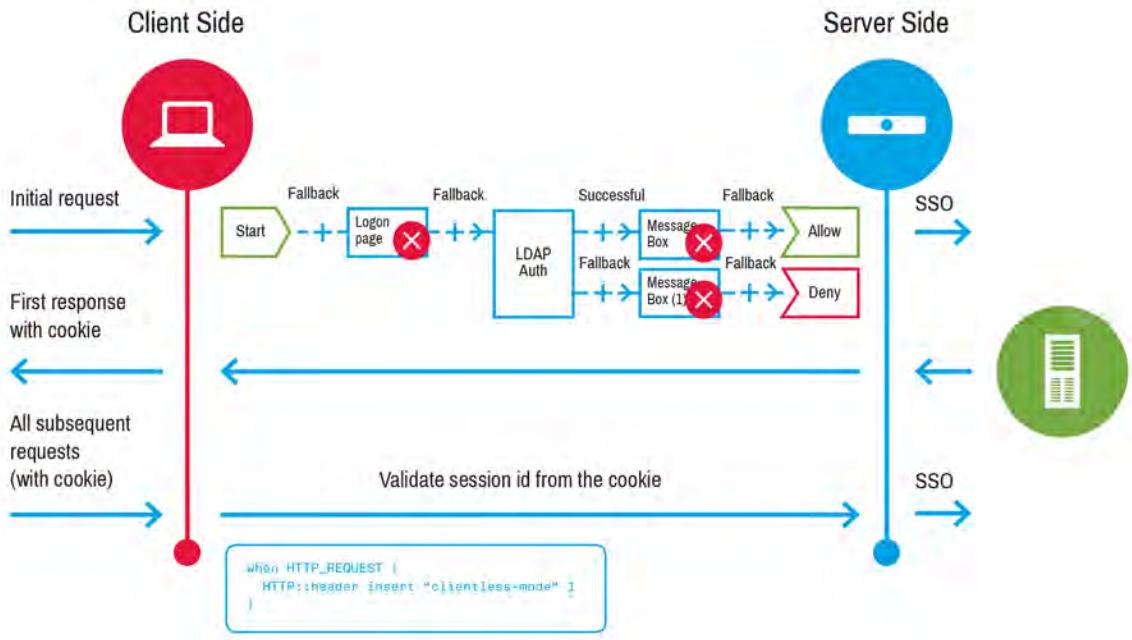

Figure 8.19: Clientless mode protocol flow

Use iRules to enable clientless mode

Clientless mode can be enabled with an iRule by inserting a “clientless-mode” header with the HTTP::header command.

```
when HTTP_REQUEST {
    HTTP::header insert "clientless-mode" 1
}
```

Clientless mode limitations

Clientless-mode does not support the following:

- **Logon Page** retry.
- Password change.
- Two factor (challenge/response based) authentications.
- Multi-domain SSO.
- On-demand client certificate.

In a situation where the clientless mode client must present credentials to gain access (such as a certificate or username/password) the HTTP_REQUEST event must be configured to happens before the access session starts. The first **HTTP_REQUEST** event will not contain an **MRHSession cookie**. For client certificate, select the request or require option in the **Client Authentication** section of the client SSL profile.

Troubleshooting

Introduction

Network access issues

Application tunnel issues

Authentication issues

Web access management issues

Portal access issues

Per application VPN issues

Single sign-on issues

Tools and utilities

Introduction

This chapter briefly covers troubleshooting methods for several of the most commonly reported issues with BIG-IP APM and includes references to existing support documentation for detailed procedures and information.

If your issue is not included, you can check other F5 self-help methods covered in the *Optimize the Support Experience* chapter in this guide. If there are configurations or issues you would like to see covered in future versions of this guide, send your detailed request by email to opsguide@f5.com.

Configuration and compatibility checks

Start with the following checks when troubleshooting your BIG-IP APM system.

NTP and DNS

Make sure that DNS and NTP are working as intended. These functions are critical to proper operation of authentication, high-availablity synchronization, and other services in BIG-IP APM.

For more information, see *General Configuration Properties* in ***BIG-IP System: Essentials.***

Client compatibility

Make first that the client in use is supported for the version of BIG-IP APM you have installed.

To find information regarding supported clients for each version of BIG-IP APM, see the *Client Compabability Matrix* for your version. Go to the ***BIG-IP APM support*** page (https://support.f5.com/kb/en-us/products/big-ip_apm.html) and select your version from the **Product Documentation** menu. On the Documentation page, under **Getting Started, Upgrades, and Reference**, click BIG-IP APM Client Compatibility Matrix.

Note If your version is a point release (11.5.2, for example), a Client Compatibility Matrix may not be available. If not, look on the support page for the previous version.

Connectivity checks

Continue general troubleshooting by checking connectivity.

Data and control plane

When testing connectivity of your BIG-IP system, keep in mind that traffic passes through two planes of the BIG-IP APM system:

- Data plane (TMM)
- Control plane (Linux)

This means that information passing through the control plane may not necessarily reflect that which passes through the data plane. For example, a ping to a server-connected VLAN from the **Configuration utility** uses the control plane. The control plane may have access to the server, but if there is no connectivity from the data plane, application traffic will fail.

F5 recommends using **tcpdump** or a BIG-IP LTM monitor to verify that the BIG-IP data plane has the necessary and correct routing information.

Virtual server connectivity

Make sure that the initial client requests arrive at the virtual server. One way to confirm this is to check a BIG-IP session report for username or client IP of the local client.

To run a session report using the Configuration utility

- Go to **Access Policy** > **Reports** > **Run Report**.

If client requests are not arriving at the virtual server, see *Virtual Server Troubleshooting* in ***F5 Local Traffic Manager and Global Traffic Manager Operations Guide***.

Session timeouts

Some issues can be caused by session timeouts. A BIG-IP APM session can terminate for various reasons, including a timeout. Sessions can time out before the access policy completes, based on the **Access Policy Timeout** specified by the policy configuration.

Sessions can also time out if they exceed a specified total session lifetime as defined by **Maximum Session Timeout**.

Default timeout settings are automatically configured for each access profile. These settings can be modified.

To modify timeout settings in the Configuration utility

1. Go to **Access Policy** > **Access Profiles.**
2. Select the profile to be modified.
3. In Settings, modify timeout values.
4. Click **Update**.

Log file checks

Log files can be used to troubleshoot access policy evaluation.

You can view log messages about the access policy evaluation as it follows each policy agent and event.

To view the BIG-IP APM log messages from the command line

- Type the following command:

```
tail -f /var/log/apm
```

Tip BIG-IP APM log messages are located in the **/var/log/apm** file.

Change logging levels

To use log messages for troubleshooting, you may need to change the BIG-IP APM log level to debug. This setting provides more complete messages.

Important Logging at the debug level creates large log files which may lower performance. Once you are finished troubleshooting, you will need to return the logging level (**Notice**).

To enable additional Access Policy logging using the Configuration utility

1. Go to **System** > **Logs** > **Configuration** > **Options** > **Access Policy**.
2. In the **Access Policy Logging** section, for **Access Policy, SSO, Portal Access,** and **VDI,** select **Debug**.
3. Click **Update.**

Tip To return to default logging level, set each item to **Notice**.

To enable additional logging using tmsh at the command line

- Type the following commands:

```
tmsh modify sys db log.accesscontrol.level { value "debug" }
tmsh modify sys db log.sso.level { value "debug" }
tmsh modify sys db log.webapplications.level { value "debug" }
tmsh modify sys db log.vdi.level { value "debug" }
```

To disable additional logging using tmsh at the command line

- Type the following commands:

```
tmsh modify sys db log.accesscontrol.level { value "notice" }
tmsh modify sys db log.sso.level { value "notice" }
tmsh modify sys db log.webapplications.level { value "notice" }
tmsh modify sys db log.vdi.level { value "notice" }
```

Access policy evaluation log messages

As an example, the following figure shows the Visual Policy Editor view of an access policy applied to a BIG-IP virtual server.

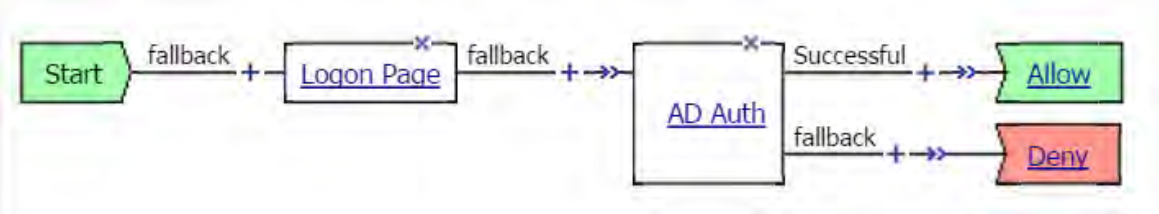

Figure 9.1 Access policy using Logon Page and AD Auth policy agents.

Tip Each log message contains a session ID, which you can use to identify a particular session when filtering large log files.

When a user first connects to the virtual server with the access policy shown in the previous figure, a new BIG-IP APM session log message is created, which will appear similar to the following:

```
 notice tmm[23456]: 01490500:5: 4da2e357: New session from client IP 172.17.2.157
(ST=/CC=/C=) at VIP 172.24.102.83 Listener /Common/testvip
```

In this example, the session is assigned session ID **4da2e357**.

Following the policy from **Start**, the next log message in the BIG-APM log file will appear similar to the following:

```
 info apd[1234]: 01490006:6: 4da2e357: Following rule 'fallback' from item 'Start' to
item 'Logon Page'
```

This message shows that the user was presented a BIG-IP APM login page by the **Logon Page** agent.

If the user attempted to log in, log messages similar to the following appear:

```
 notice apd[1234]: 01490010:5: 4da2e357: Username 'administrator'
 info apd[1234]: 01490006:6: 4da2e357: Following rule 'fallback' from item 'Logon Page'
to item 'AD Auth'
 info apd[1234]: 01490005:5: 4da2e357: Following rule 'Successful' from item 'AD Auth' to
ending 'Allow'
```

These three log messages show the user **administrator** was successfully authenticated with BIG-IP APM and allowed access to the target resource:

- The first log message shows that the BIG-IP APM system received a login attempt from user with the username **administrator**.
- The second log message shows that the policy followed the **fallback** branch to the **Logon Page** policy agent.
  That agent received valid information and the access policy then followed the **fallback** branch to the **AD Auth** agent.
  The AD Auth agent in this example took the username and password stored in the session cache and verified it against an Active Directory AAA resource configured in it.
- The third log message shows **AD Auth** agent executed successfully. This completes the access policy evaluation. The path followed the **Successful** branch and terminated at **Allow**.

Network access issues

When issues with BIG-IP APM network access arise, the first thing to do is check the BIG-IP APM log messages to see whether or not network access resource at issue has been correctly assigned and started.

Tip BIG-IP APM log messages are located in the **/var/log/apm** file.

Log message troubleshooting

The following list shows sample message logs indicating successful evaluation of various network access resources actions. Messages on your system will vary but should look similar to these.

- **Log message indicating network resource was successfully assigned**:

```
 notice apd[12025]: 01490008:5: e0401776: Connectivity resource
'/Common/test_network_access' assigned
```

- **Log message indicating that an IP address has been assigned:**

```
 notice tmm3[16234]: 01490549:5: e0401776: Assigned PPP Dynamic IPv4: 10.40.10.9
Tunnel Type: VPN_TUNNELTYPE_TLS NA Resource
```

- **Log message indicating that the PPP tunnel was correctly started:**

```
 notice tmm3[16234]: 01490505:5: e0401776: PPP tunnel 0x5700dd101400 started.
```

- **Log message confirming that the maximum number of access sessions has not been reached:**

```
 warning tmm[16234]: 01490509:4: d8330ccb: Concurrent user sessions limit reached
for access profile: /Common/test_access_policy
```

Note The **Max Sessions Per User** setting in an access profile will periodically terminate older sessions.

If your logs show no errors and the client is not able to establish a tunnel, continue troubleshooting from the perspective of the client.

Client-side troubleshooting

Note These troubleshooting options are only available on Windows client systems.

The following procedures check common issues associated with establishing network access and the Windows BIG-IP Edge Client from the client perspective.

Privilege troubleshooting

Several BIG-IP Edge Client components require elevated privileges to install and function properly. If a user without local administrator privileges is experiencing an issue, you can test whether the privileges assignment is the problem.

To test whether or not privileges are the cause of an issue

1. Log in to the client system experiencing the trouble using an account with local administrator privileges.

2. Attempt to reproduce the issue on the client system.

If you cannot reproduce the issue, it is possible that the cause is insufficient Windows privileges for non-administrator accounts.

To test administrative privileges

1. Assign local administrator privileges to the user's account on the client system.

2. Install all required BIG-IP Edge Client components and the component installer service on the client system.

If the components do not install correctly, the problem may not be a privileges issue but may originate from the user's machine.

To test whether the issue is machine-related

1. Locate another machine on the same Windows domain as the one having the issue.
2. Log in using the account of the user experiencing the network access issue.
   If you cannot reproduce the issue, it is likely the issue originates from the user's machine.
   If you can reproduce the issue, try again on multiple machines in the same Windows domain. If you can reproduce the issue on multiple machines, it is likely the issue orignates from BIG-IP Edge Client software.

Important: Windows client systems that are members of a Windows domain may have a group policy which affects the operation of the network access (VPN) connection. F5 recommends that you attempt to reproduce the issue using at least one system that is *not* a member of the same Windows domain.

Network connectivity troubleshooting

If you have followed the procedures previously listed and the user is still experiencing network activty issues, check network connectivity with the following methods:

- Check TCP/IP connectivity on the client system using tools such as the **ping** utility.
- Visit the BIG-IP APM login page using several different web browsers to make sure the issue isn't related to the client browser.
- Run the **nslookup** or **dig** utilities on the client system to verify that DNS is resolving properly. The BIG-IP APM hostname must resolve to only a single IP address, so make sure that multiple records are not returned.

BIG-IP Edge Client troubleshooting

If you have followed the procedures previously listed and the user is still experiencing network activity issues, remove and then re-install all BIG-IP Edge Client components on the client system using the **component troubleshooting** utility.

For more information about removing BIG-IP Edge Client components, see AskF5 article ***SOL8253: Removing BIG-IP APM and FirePass client components from Windows client systems***.

Configuration items troubleshooting

If you have followed the procedures previously listed and the user is still experiencing network activity issues, using the **Configuration utility**, check the following settings for the network access resource in question:

1. Make sure **Traffic Options** is set to **Use split tunnelling for traffic**.
2. Make sure SNAT address can be reached when **SNAT Pool** is set to **Auto Map**.
3. If VPN clients use IP addresses from the LAN IP subnet, make sure **Proxy Arp** is enabled.
4. Make sure that the domain name setting for **DNS Address Space** is correct.
5. Make sure that no ACLs assigned to the network resource are blocking access to a back-end resource.

Note The **tcpdump** utility tool can be used to diagnose protocol errors and DNS and/or routing problems. For more information, see Ask F5 article ***SOL411: Overview of packet tracing with the tcpdump utility***.

Application tunnel issues

Application tunnels may fail to establish for the following reasons:

- DNS resolution fails for the application tunnel resource host.
- The access policy fails to assign the resource.
- Application launch is incompatible with the client OS.
- Connectivity issues prevent back-end resources from being reached.

DNS resolution troubleshooting

When an application tunnel resource is configured using a hostname, BIG-IP APM tries to resolve the hostname for the configured host. If DNS resolution fails, check the BIG-IP APM log file. If hostname cannot be resolved, log messages similar to the may appear:

```
 err apd[12025]: 0149015d:3: 695b8701: DNS request to resolve hostname
(somehostthatdoesnotexist.corp) failed for app tunnel resource (/Common/myapptunnel).
errno(139717604) h_errno(-2) Error str(Name or service not known)
 err apd[12025]: 01490000:3: modules/EndingAgents/Allow/AllowAgent.cpp func:
"createATImplicitACL()" line: 695 Msg: Remove AT(/Common/myapptunnel) as DNS resolution
failed for somehostthatdoesnotexist.corp
```

Tip BIG-IP APM log messages are located in the **/var/log/apm** file.

You can also confirm the BIG-IP system DNS configuration. The **nslookup** command line utility can be used to determine whether or not the configured host can be resolved by the BIG-IP system.

Access policy assignment troubleshooting

If the access policy fails to assign a resource to a user, check the BIG-IP APM log file. A successful assignment of an application tunnel resource is logged in the BIG-IP APM log file. A log message similar to the following may appear:

```
apd[12025]: 01490008:5: 695b8701: Connectivity resource '/Common/myapptunnel' assigned
```

You can also review the access policy to make sure that the user is assigned the correct group or user privlieges to have access to the resource.

Application launch incompatibility troubleshooting

To access an internal hostname using application launch with an application tunnel resource, specify Application Path parameters associated with the application.You can add the following parameters:

- **%host%** Substituted with the loopback host address. For example: **http://%host%/application/**.
- **%port%** Loopback port. Use this if the original local port has changed due to conflicts with other software.

If these parameters are left out, the application launch may fail. F5 recommends installing BIG-IP DNS Relay Proxy service to make sure that resolution of all internal hostnames succeeds.

For more information see AskF5 article: ***SOL9694: Overview of the Windows DNS Relay Proxy service***.

Connectivity troubleshooting

The BIG-IP APM needs to be able to connect to back-end servers for an application tunnel to succeed. Linux utilities such as **ping** and **curl** can be used to confirm that back-end resource addresses are accessible from BIG-IP APM.

If user sessions are configured to use SNAT settings, confirm that the SNAT address is routable to the VLAN on which the destination resource is located.

An assigned ACL may block access to back-end resources. ACLs are evaluated in numerical order, from lowest to highest. Check to make sure an ACL isn't responsible for blocking access.

If protocol errors and/or routing errors prevent the application tunnel from establishing, the **tcpdump** utility can be used to capture traffic for detailed analysis.

For more information, see AskF5 article: ***SOL411: Overview of packet tracing with the tcpdump utility***.

Authentication issues

If users cannot authenticate during access policy evaluation, troubleshoot the authentication (AAA) elements listed in this section, as appropriate for the resource types.

Note Complete troubleshooting described in the previous sections of this chapter, on both the client side and server side of your BIG-IP APM system before troubleshooting authentication issues.

Active Directory troubleshooting

Active Directory AAA resources use the Kerberos protocol for communicating with the Active Directory server(s). If users are experiencing an authentication issue, make sure the following network ports are not blocked between the BIG-IP APM and Active Directory server:

- TCP/UDP port 88
- TCP/UDP port 464
- TCP/UDP port 53

If the issue persists, continue troubleshooting by making sure of the following:

- NTP is configured on the BIG-IP APM and there is no more than a 5-minute difference between the clocks on BIG-IP APM and the Active Directory servers.
- DNS is configured correctly on BIG-IP APM.
- DNS SRV records are correctly returned for Active Directory domain controllers.
- **Split Domain From Username** option is enabled in the Logon Page agent appearing before the AD Auth agent in the access policy.

Tip For troubleshooting purposes, you can enable **Show Extended Error** in the AD Auth policy agent. This setting returns a full error in the case of authentication failure.

In Active Directory environments that have domain trusts or parent-child domain relationships, enable **Cross Domain Support** in the AD Auth policy agent in the resource access policy. This setting makes sure Kerberos referrals can be followed to child or other domains in the AD forest.

For more information on Active Directory troubleshooting, see the following AskF5 articles:

- ***SOL11628: Error Message: Clock skew too great, principal name: [username]***
- ***SOL11473: Error Message: Client not found in Kerberos database***
- ***SOL11830: Error Message: Server not found in Kerberos database***

RADIUS troubleshooting

The RADIUS protocol provides access control for network devices using one or more centralized servers. RADIUS operates over UDP and provides AAA management for users connecting to a network service.

RADIUS messages sent between the BIG-IP APM and RADIUS server are authenticated through the use of a shared secret.

When troubleshooting issues with RADIUS make sure of the following:

- UDP port 1812 between BIG-IP APM and the RADIUS server(s) is not blocked.
- The shared secret matches on BIG-IP APM and the RADIUS server(s).
- Network availability of the remote RADIUS server is availailable. Use a utility, such as **ping** or **traceroute**.

Trace RADIUS traffic

To troubleshoot RADIUS sessions, you can use **tcpdump** packet to capture traffic.

To packet trace RADIUS traffic at the command line using tcpdump

- If the RADIUS server is reachable on the management network, type the following syntax:

```
tcpdump -s0 -ni eth0 port 1812 -vw /shared/tmp/rad.pcap
```

- If the RADIUS server is reachable on a TMM network, type the following syntax:

```
tcpdump -s0 -ni <vlan_name> port 1812 -vw /shared/tmp/rad.pcap
```

Note The **-vw** switches write the output to **/shared/tmp/rad.pcap**. It also starts a packet counter to show if any packets are written to the file.

Once the traffic is captured, you can view the capture file using a packet analysis program.

For more information on troubleshooting RADIUS, see AskF5 article: ***SOL15789: Troubleshooting RADIUS authentication for application traffic***.

RSA SecurID troubleshooting

When troubleshooting system-wide issues authenticating against RSA SecurID, make sure of the following:

- The IP address of the RSA SecurID server in the BIG-IP APM configuration.
- TCP port 1645 between BIG-IP APM and the RSA SecurID server is not blocked.
- There is no more than a 5-minute difference between the clocks on BIG-IP APM and the RSA SecurID server.
- The **Agent Host IP Address** matches the agent host record on the RSA SecurID server.

For more information, see the following AskF5 articles:

- ***SOL12117: Overview of the Agent Host IP Address setting for Native RSA SecurID authentication***
- ***SOL12164: Troubleshooting RSA SecurID authentication***

LDAP troubleshooting

The BIG-IP APM system supports LDAP as an authentication method. If users experience authentication issues with LDAP authentication, troubleshoot the communication between the LDAP server and the BIG-IP system to find the cause of the failure.

When troubleshooting LDAP issues, make sure of the following:

- If using an LDAP pool, the IP address of LDAP server(s) is correct.
- TCP port 389 (LDAP) is open between BIG-IP APM and the LDAP server.
- TCP port 636 (LDAPS) is open between BIG-IP APM and the LDAP server or pool members.
- LDAP server or pool members can resolve using DNS.
- LDAP administrator credentials are correct.

For more information, see the following AskF5 article: ***SOL13328**: Troubleshooting LDAP authentication with tcpdump*.

Kerberos troubleshooting

A Kerberos AAA resource is used to validate Kerberos tickets provided by users and to provide authenticated access to back-end resources through the BIG-IP APM.

If users experience issues logging in to the BIG-IP APM system using Kerberos authentication, troubleshoot the communication between the key distribution center (KDC) server and the BIG-IP system.

When troubleshooting Kerberos issues, make sure of the following:

- Key distribution center (KDC) is online.
- Network connectivity from the BIG-IP system to the KDC functions.
- TCP and UDP ports 88 (Kerberos) are open between the BIG-IP system and the KDC.
- The KDC can be resolved using DNS.

If you want to verify Kerberos authentication configurations, use the following procedures:

To verify the keytab for client-side Kerberos

- From the command line, type the following *klist* command syntax:

```
klist -kt <path to keytab file>
```

Example:

```
klist -kt /config/filestore/files_d/Common_d/kerberos_keytab_file_d/\:Common\:SUN-SPNEGO-APM106_key_file_2
```

Important: The complete command must be typed on one line.

The output will appear similar to the following:

```
 Keytab name:
FILE:/config/filestore/files_d/Common_d/kerberos_keytab_file_d/:Common:SUN-SPNEGO-
APM106_key_file_2 KVNO Principal
```

```
3      HTTP/apm106.labt.companynet.com@labt.companynet.com(arcfour-hmac)
```

For more information on troubleshooting Kerberos, see the following AskF5 articles:

- ***SOL11830: Error Message: Server not found in Kerberos database***
- ***SOL11474: Error Message: Cannot contact any KDC for realm***

Online certificate status protocol responder troubleshooting

The Online Certificate Status Protocol (OCSP) enables applications to determine the revocation status of a certificate. An OCSP client, in this case the BIG-IP APM, acts as the client, and issues a status request to an OCSP responder and suspends acceptance of that certificate until the responder provides a response.

If your users are experiencing authentication issues and you are validating client certificates with an OSCP responder, troubleshoot the resopnder.

When troubleshooting OCSP responder issues, make sure of the following:

- OCSP URL is configured correctly in the OCSP AAA resource.
- BIG-IP APM can resolve the OCSP URL
- The access policy is configured to request a client certifcate through either--but not both--the **On-Demand Cert Auth** agent or the client SSL profile.
- OCSP AAA is configured with a CA certificate, CA certificate bundle, or validation authority (VA) certifcate to validate the signature of the returned response.

The following list contains some common error messages and their descriptions, simiar to those which may appear in the BIG-IP APM log file.

- **The OCSP URL is missing in the OCSP Auth policy agent configuration:**

```
warning apd[29223]: 01490146:4: e006090f: OCSP Auth agent: Failure status 'Bad
param in module configuration'
```

- **OCSP responsder is not reachable:**

```
warning apd[29223]: 01490146:4: e7bd4ddf: OCSP Auth agent: Failure status 'Failed
to connect to OCSP responder host (192.168.49.68) at port 8888'
```

- **BIG-IP APM is not able to find the correct CA certificate for the client certificate:**

```
warning apd[12557]: 01490146:4: a7d52582: OCSP Auth agent: Failure status 'Issuer
certificate not found for the session'
```

- **BIG-IP APM did not receive a client certificate:**

```
warning apd[12557]: 01490146:4: b4577458: OCSP Auth agent: Failure status
'Certificate not found for the session'
```

- **BIG-IP APM is unable to validate the OCSP responder's certificate chain:**

```
warning apd[29223]: 01490146:4: 4717a222: OCSP Auth agent: Failure status
'Response Verify Failure'
```

For more information on troubleshooting OCSP, see *OSCP Authentication* in ***BIG-IP Access Policy Manager: Authentication and Single Sign-On***.

Certificate revocation list distribution point troubleshooting

BIG-IP APM supports retrieving certificate revocation lists (CRLs) from network locations/distribution points (DP). A certificate revocation list distribution point (CRLDP) AAA server defines how BIG-IP APM accesses a CRL file from a distribution point. A distribution point is either an LDAP URI—a directory path that identifies the location where the CRLs are published—or a fully qualified HTTP URL.

CRLDP is used to validate an SSL client certificate for authentication purposes during access policy execution. If you experience issues logging in to the BIG-IP APM system and are using CRLDP authentication, troubleshoot the communication between the CRLDP distribution point and the BIG-IP system to determine the cause of the failure.

When troubleshooting CRLDP issues, make sure of the following:

- CRLDP is online.
- Network connectivity from the BIG-IP APM system to the CRLDP is functioning.
- A valid CRLDP AAA is assigned to the **CRLDP** agent in the access policy.
- The access policy is configured to request a client certifcate through either—but not both—the **On-Demand Cert Auth** agent or the client SSL profile.

For more information on troubleshooting CRLDP, see *CRLDP Authentication* in ***BIG-IP Access Policy Manager: Authentication and Single Sign-On***.

Web access management issues

Web access management issues typically involve configuration of virtual servers, HTTP profiles, or access policies, or connectivity problems.

To determine whether your issue is related to BIG-IP LTM configuration or BIG-IP APM access policy configuration, remove the access profile from the virtual server and see if you can reproduce the problem.

If the issue still occurs, then the problem is likely caused by a connectivity issue or an issue with the configuration of the virtual server or HTTP profile.

If the issue no longer occurs, troubleshoot the access policy.

Virtual server troubleshooting

When troubleshooting virtual server issues, make sure of the following:

- Connectivity between the client and the BIG-IP APM system is functioning.
- IP connectivity between the BIG-IP system and any dependent resources, such as pool members, is functioning.
- SNAT settings are correct.
- A pool is associated with the virtual server.
- At least one of the pool members is available.
- HTTP profile settings are correct.
- Clientssl profile and/or Serverssl profile settings are correct.

Access policy troubleshooting

When troubleshooting access policy issues, make sure of the following:

- Correct endings exist for all access policy branches.
- Correct results return after access policy execution. (For example, the user's access device is correct when using the **Client Type** policy agent.)

Access policy execution log messages troubleshooting

Log messages showing a successful access policy evaluation will look similar to the following:

```
 notice tmm[23456]: 01490500:5: 4da2e357: New session from client IP 172.17.2.157
(ST=/CC=/C=) at VIP 172.24.102.83 Listener /Common/testvip
 notice apd[1234]: 01490010:5: 4da2e357: Username 'administrator'
 info apd[1234]: 01490006:6: 4da2e357: Following rule 'fallback' from item 'Logon Page'
to item 'AD Auth'
 info apd[1234]: 01490005:5: 4da2e357: Following rule 'Successful' from item 'AD Auth' to
ending 'Allow'
 notice apmd[6261]: 01490102:5: /Common/swg-explicit-ap:Common:cae7b0d8: Access policy
result: LTM+APM
```

Tip BIG-IP APM log messages are located in the **/var/log/apm** file.

If the log messages show *unsuccessful* policy evaluation, look for authentication errors or configuration problems in the access policy.

If log messages show *successful* policy evaluation but the issue persists, check to see if any assigned ACLs are blocking access to back-end resources.

Portal access issues

If a portal access resource is not working as expected, begin troubleshooting by verifying the access policy executes as expected and the appropriate resource is assigned.

Several browser plugins and tools exist for capturing HTTP traces, such as HTTPWatch, Fiddler and the Developer Tools for Google Chrome, Mozilla Firefox, and OS X Safari browsers.

HTTPWatch's HWL file or an HTTP Archive (HAR) file are F5 preferred formats.

Note: Clear the client's browser cookies and cache, then close and re-open the browser before capturing a new HTTP trace.

If users are experiencing portal access issues, troubleshoot the communication between the BIG-IP APM and the configured resource items to determine the cause of the failure.

When troubleshooting portal access issues, make sure of the following:

- IP addresses of resource items are correct.
- The defined paths for the resource items allow access to the correct URI Path for the rewritten application.
- If enabled, the **Match Case for Paths** setting is not causing URL pattern match issues.
- **Scheme** and port for the resource item is correct.
- **Client Cache** settings are not causing stale information to be sent to clients.

Browser incompatibility and script troubleshooting

If the portal access issues settings are not contributing to the issue, continue troubleshooting by doing the following:

- Rule out browser incompatibilty and script issues.
- Capture an HTTP trace from the client directly to the web application.
- Capture an HTTP trace from the client using the portal access resource.

To determine if browser incompatibility exists, use a different web browser to connect to the resource. If the issue no longer occurs, there may be a incompatibility issue with the previous web browser or web browser version.

Make sure that the portal access application does not contain any script errors. Attempt to execute the problematic function with script errors enabled, or by viewing the web console.

The list below shows the procedure for accessing the web console on the current versions of the three most popular browsers:

- Internet Explorer: **Tools > Internet Options > Advanced > Display a notification about every script error**.
- Chrome: **Customize and Control Google Chrome > More Tools > Developer Tools > Console**.
- Firefox: **Open Menu > Developer Tools > Toggle Tools > Console**.

To get the best user experience with a portal access resource, the web application must execute with no errors. Warnings are acceptable, but if errors should be addressed within the browser as they may be the cause of the issue.

Use HTTP trace from client direct to web application in troubleshooting

HTTP traces taken from the client directly to the back-end application without BIG-IP APM provide a picture of what a successful request or HTTP session should look like.

When capturing the HTTP trace, the client must use the server hostname (http://example.com/) and not IP address (http://192.168.1.100/). The HTTP trace must be started before the first request is executed.

Use HTTP trace through a BIG-IP APM portal access resource in troubleshooting

An HTTP trace taken while accessing a portal access resource on the BIG-IP APM will show the application failure.

The HTTP trace **must be started before** the first request.

For additional troubleshooting tools, see the AskF5 article: ***SOL13384: Performing a web applications trace (11.x)*.**

Per-application VPN issues

Per-application (Per-App) VPN problems are typically caused by one of the following:

- MDM policy issues
- Access Policy issues
- Connectivity issues

MDM policy troubleshooting

Per-App VPNs require deployment of a third-party mobile device management (MDM) application. Ab MDM policy must be correctly applied to mobile device. Without this policy, Per-App VPN tunnels are cannot be created. If users experience issues with Per-App VPNs and the BIG-IP APM, troubleshoot the MDM solution.

When troubleshooting MDM issues, make sure of the following:

- VPN profile settings are correct.
- MDM profile settings are correct.

There are two limitations for applications that use iOS 7 Per-App VPN API:

- Only managed applications installed on the user's device using an MDM solution are able to use iOS 7 Per-App VPN.
- Only TCP applications are currently supported.

Important: Apple's current implementation of the iOS 7 Per-App VPN API are the source of these limitations and are subject to change in any future iOS release.

Access policy troubleshooting

The BIG-IP APM log file should be reviewed to make sure the access policy is successful. If it is not, review the message logs for any error messages related to:

- Authentication errors.
- Misconfiguration of access policy items .

Additionally, review the BIG-IP APM log file to make sure of the following:

- Correct endings exits for all access policy branches.
- Correct result is returned from each access policy agent. (For example, the user's access device is correct when using the **Client Type** agent.)

Tip BIG-IP APM log messages are located in the **/var/log/apm** file.

If the access policy completes successfully, confirm that no assigned ACLs are is blocking access to the back-end resources.

Connectivity troubleshooting

If the access policy starts and completes successfully, the problem is likely a connectivity issue. The BIG-IP APM must be able to connect to the backend server for the per-app VPN to succeed.

When troubleshooting connectivity issues, make sure of the following:

- Connectivity between the client and the BIG-IP system is functioning.
- IP connectivity between the BIG-IP system and any dependent resources, such as internal hosts, is functioning.
- SNAT settings are correct on the BIG-IP LTM vitual server.

Single sign-on issues

Symptoms of issues with single sign-on (SSO) include the following:

- Application prompts users to sign more than once.
- Application loops on the login page.
- Invalid login message appears on the login page.
- SSO is successful the first time but fails on subequent attempts.

Common issues troubleshooting

SSO mechanisms in BIG-IP APM require separate troubleshooting techniques, but there are several common causes of issues. When you begin troubleshooting SSO, you can start by seeing if any of the following are the cause of the issue:

- Once an SSO failure is detected within a user session, the SSO becomes disabled for all SSO types.
- SSO mechanisms that require a password read the access session variables created by the **SSO Credential Mapping** policy agent. This agent must execute during access policy execution for each user session.
- HTTP NTLM and Kerberos SSO methods require correct configuration of the session **session.logon.last.domain** variable. The variable can be set manually with a **Variable Assign** policy agent. It is also assigned automatically if users provide a domain in the **Logon Page** policy agent and log in using a **domain\username** or **username@domain** format.
- Kerberos SSO (S4U) and SAML SSO types do not require a password.
- Kerberos AAA, NTLM (ECA), and client certificate (OCSP, CRLDP, or on-demand) AAA types do not provide a password. SecurID RSA AAA (or RADIUS with RSA) AAA types typically provide an OTP token that does not represent a user password but has validity for internal resources.

For more information, see AskF5 article: ***SOL13595: Frequently used tools for troubleshooting BIG-IP APM and Edge Gateway issues (11.x)***.

Forms (server-initiated) troubleshooting

With forms SSO, the Web SSO process transmits user credentials to the internal web server when the user accesses the **Start URI**.

To troubleshoot forms SSO

1. Increase log verbosity, as described in *Increase log verbosity* in this section.
2. Use a Linux text utility, such as **less** or **tail** to follow the BIG-IP APM log messages.
3. Log in as a test user and attempt SSO processes.
4. Review log messages. See the following procedure for more information.

Tip BIG-IP APM log messages are located in the **/var/log/apm** file.

When troubleshooting SSO, make sure of the following, in the order listed:

1. The correct **Forms SSO** policy agent is used in the access policy.
2. Client issues its request to the correct **Start** URI, configured in the **Forms SSO** access policy agent.
   Web SSO logs the **Start** URI and the request URI. These must match exactly so that the **Forms SSO** agent is triggered.
   In the following log message example, SSO is configured with a **Start** URI of **mywebapp.corp/login.asp**, and the client's request an unmatched URI of **"/"**. The resulting output is **no start uri match**.

   ```
    debug websso.0[22225]: 014d0030:7: 4e763e75: checking start uri match, start uri:
   '/mywebapp.corp/login.asp', request:"/"
     debug websso.0[22225]: 014d0030:7: 4e763e75: no start uri match
   ```

3. **Successful Logon Detection Match Type** and **Successful Logon Detection Match Value** are selected are working correctly.
   If these options are not correctly configured, SSO may be inadvertently disabled. The following sample syntax shows this condition:

   ```
    notice websso[16111]: 014d0038:5: c2bd29cd: success match failed for 'test' using
   config '/Common/myformssso', SSO disabled for this session
   ```

Forms-client initiated troubleshooting

With forms client-initiated SSO, the Web SSO process inserts a script into the web app's login page. This script causes the client to automatically POST credentials to the web app, mimicking a user login.

To troubleshoot forms-client initiated SSO

1. Set SSO logging to debug level within the **Forms-Client Initiated SSO** policy agent.
2. Use a Linux text utility such as **less** or **tail** to follow the BIG-IP APM log messages.
3. Use an HTTP trace utility with recording capability, such as Fiddler, HTTPWatch, or HTTPFox to record a web session on the test user's local machine.
4. Enable the web browser's JavaSscript error console to make visible any JavaSscript errors.
5. Log in as a test user and attempt SSO.
6. Review the log messages and recorded session. See the following procedure for more information.

Important Set log verbosity for the **Forms-Client Initiated SSO** policy agent within the policy agent object rather than as described in *Increase log verbosity*.

When reviewing log messages and web session recording, make sure of the following, in the order listed:

1. The correct **Forms Client-Initiated SSO** policy agent is used in the access policy by looking for a log message with a syntax similar to the following.

```
 Mar 16 23:31:58 f5b info tmm1[19390]: 014d0002:6: 996e8bee: SSOv2 Request "GET /",
config /Common/my_client_initiated_sso
```

The config **/Common/my_client_initiated_sso** portion of the previous sample message indicates the **Forms Client-Initiated SSO** agent is being used.

2. Client issues its request to the correct **Start** URI, configured in the **Forms Client-Initiated SSO** access policy agent. Also check that the form is detected correctly. Do this by looking for log messages with the synax similar to the following:

```
 info tmm1[19390]: 014d0002:6: 996e8bee: SSOv2 Request match, config
/Common/my_client_initiated_sso form Loginform info tmm1[19390]: 014d0002:6:
996e8bee: SSOv2 Form detected, config /Common/my_client_initiated_sso form
Loginform
```

   The **Request match** and **Form detected** portions of the previous sample log messages indicate correct configuration.
3. In the recorded session data, the HTML body of the web app's login page includes JavaSscript near the bottom to automatically POST the credentials. It can be identified by the text **f5-sso-token**, which is the password transmitted by the client.
4. No script errors occur on the client web browser.

Note Some web apps use script attached to the form submit button or form onsubmit action to perform various tasks. In such cases, the inserted JavaScript must be customized specifically for the web app's login page. This procedure requires JavaScript programming experience and is outside the scope of this document. For further assistance, contact ***F5 Professional Services*** (https://f5.com/support/professional-services).

NTLMv1 SSO, NTLMv2 SSO, and HTTP basic SSO troubleshooting

NTLMv1 SSO, NTLMv2 SSO, and HTTP basic SSO methods use standard HTTP authentication mechanisms transmitted on the user's behalf by BIG-IP APM. The authentication will appear within the internal server traffic.

To troubleshoot NTLMv1 SSO, NTLMv2 SSO, and HTTP basic SSO

1. Increase log verbosity, as described in *Increase log verbosity* in this section.
2. Use a Linux text utility such as **less** or **tail** to follow the BIG-IP APM log messages.
3. Use **ssldump** to decrypt HTTPS or **tcpdump** to capture HTTP communication between BIG-IP APM and the internal web server.
4. Log in as a test user and attempt SSO.

When reviewing log messages and decryption/capture information, make sure of the following, in the order listed:

1. Recorded HTTP transactions for HTTP 401 responses from the internal server are correct.
2. The correct SSO profile is used for the session. Check this by looking for a syntax similar to the following:

   ```
    info websso.3[27034]: 014d0014:6: 52bd552a: Found HTTP 401 response for SSO
   configuration '/Common/my-http-sso' type:'ntlmv1'
   ```

   The **/Common/my-http-sso' type:'ntlmv1** portion indicates the SSO profile used.
3. The internal server Web SSO process returns a "Found HTTP 401" response. Check this by looking for syntax similar to the following:

   ```
    info websso[27034]: 014d0014:6: 52bd552a: Found HTTP 401 response for SSO
   configuration '/Common/my-http-sso' type:'ntlmv1'
   ```

   The **Found HTTP 401 response** portion indicates the 401 response.
4. SSO is not disabled for the session. Check this by looking for syntax similar to the following:

   ```
    " _ssoDisabled: true "
   ```

For more information, see the following AskF5 articles:

- ***SOL10209: Overview of packet tracing with the ssldump utility***
- ***SOL411: Overview of packet tracing with the tcpdump utility***

Kerberos SSO troubleshooting

Kerberos SSO utilizes a Kerberos extension called Service 4 User (S4U). Web SSO obtains a Kerberos ticket on behalf of the user and either inserts it into the request sent to the server or waits for the HTTP 401 response from an internal server.

Common Kerberos failures are caused by the following:

- DNS resolution failure. For Kerberos SSO to function, a PTR record and an A record must exist and be correctly configured for target resources.
- NTP must be configured and working correctly.
- Duplicate service principal names (SPNs) must be corrected.

When troubleshooting Kerberos SSO make sure of the following:

- The SSO policy agent is correctly configured and applied in the access profile.
- Transitive trust is in place for cross-realm or cross-forest Kerberos delegation and user's realm specified in the **User Realm Source** variable. The BIG-IP APM Kerberos SSO module will not follow enterprise canonical referrals.
- Delegation account belongs to the same realm as the back-end resource.
- Delegation account is configured properly.

To verify server-side Kerberos delegation at the commmand line

- Type the following syntax:

```
kinit <SPN of delegation account>
```

  Example:

```
kinit HOST/krbsvc.example.com@EXAMPLE.COM
```

You will be prompted for a password and should receive a ticket (no output, no error). The *kinit* command retrieves a ticket-granting ticket (TGT) from the key distribution center (KDC) for the delegation account.

Server-side Kerberos delegation also performs protocol transition, or S4U2Self.

To test S4U2Self functionality at the command line

- Type the following syntax:

```
kvno -U <valid user> <SPN of delgation account>
```

  Substitute a valid AD user account and SPN of the delegation account.

  Example:

```
 kvno -U bob.user HOST/krbsvc.example.com@EXAMPLE.COM
```

A successful kvno command will return a key version number (kvno).

For more informatin on configuring DNS A, PTR, and SRV records and on resolving duplicate SPNs, search Microsoft KB articles on ***Microsoft's Support search page***. (This link takes you to an outside resource.)

Tools and utilities

Several command line tools can be used to troubleshoot the BIG-IP APM system. **sessiondump** and **configdump** utilities can be used to check current user session data while the **tcpdump** and **ssldump** utilities can be used to analyze traffic traversing the BIG-IP APM system.

Additionally, a client-based component troubleshooting utility can be used to debug logging, diagnostic reports and manual removal of the client components

sessiondump

Use **sessiondump** to view session variables for one or more active sessions, including sessionid, assigned resources profiles, and others. **sessiondump** includes the following commonly used commands:

- **-allkeys** returns all session variable for all current sessions.
- **<Session ID>** returns all session variables for the specified 8-character session id.

To see all the **sessiondump** options, type **sessiondump** at the BIG-IP APM command line and press **Enter**.

The following sample output shows **sessiondump** using the **-allkeys** option:

```
[root@sam10:Active] config # sessiondump -allkeys
04447e63 10 SessionKey
04447e63.session.access.profile 10 jm_ap_web1
04447e63.session.assigned.resource_groups 8 jm_rg_1
04447e63.session.assigned.resources 47 again google jm_na_res1 my_new_app vim webApp2
04447e63.session.assigned.uuid 17 UUID:j:jm_ap_web1
04447e63.session.assigned.webtop 18 jm_portal_web_vim1
04447e63.session.client.activex 1 0
04447e63.session.client.cpu 3 x86
04447e63.session.client.js 1 1
```

Tip the **-all keys** command in **sessiondump** can return large amounts of data on busy systems.

configdump

Use **configdump** to view configuration variables for access policies and includes the following commonly used commands:

- **-list** returns the list of configuration snapshots
- **-allkeys** returns all configuration variables for all configuration snapshots
- **<Configuration self device ID>** returns all configuration variables for the specified configuration ID.

Below is an example of a partial output using the **-list** option:

```
[root@sam10:Active] config # configdump -list
Configudump using self device name: /Common/bigip3907mgmt.lab.fp.f5net.com
Configudump using self device ID: 9d67a1fda622c7617240fa9e3ffc885a
/Common/accessSSL3.1428085704 32 282f098ade80_1ooooooooooooooooo
/Common/accessSSL3.current 10 1428085704
/Common/accessSSL3.1428085703 32 27db53b9490e2_0ooooooooooooooooo
```

tcpdump

Use **tcpdump** to trace packets in a BIG-IP APM network access tunnel by specifying the name of the connectivity profile as the interface.

For example:

```
tcpdump -i <apm_connectivity_profile_name>
```

To view the current BIG-IP APM connectivity profile using the Configuration utility

1. Go to **Local Traffic > Virtual Servers**, and click the name of the virtual server for the network access policy you want to trace.
2. The profile name is displayed in the **Connectivity Profile** menu.

For more information, see AskF5 article: ***SOL411: Overview of packet tracing with the tcpdump utility*****.**

ssldump

Use **ssldump** to examine, decrypt, and decode SSL-encrypted packet streams managed by the BIG-IP system.

For more information, see AskF5 article: ***SOL10209: Overview of packet tracing with the ssldump utility***.

Component troubleshooting utility

The **Component troubleshooting** utility can provide debug logging, diagnostic reports and manual removal of the client components. The report that the utility generates is required, in most cases, when opening support cases for client-side problems.

For more information, see AskF5 article: ***SOL12444****: Overview of the Component Troubleshooting Utility*.

Optimize the support experience

Introduction

F5 support services

Self help

Training

Engage support

Open a support case

Collect BIG-IP APM data

Share diagnostic files with F5

F5 technical support commitment

F5 strives to continuously improve its support service and create closer customer relationships. Designed to provide assistance with specific break-fix issues and ongoing maintenance of F5 products, F5 professional support services are consistently high-quality.

This means:

- F5 network support engineers conduct themselves professionally at all times.
- F5 is committed to providing the best customer experience possible.
- Customers are treated with respect and given every consideration possible.
- F5 aims to provide resolutions the first time, every time.
- Manager escalation is always available for unresolved or "site down" issues.

Some technical support issues arise from configuration errors, either within the BIG-IP system or with other devices in the network. In other cases, a misunderstanding of BIG-IP capabilities can lead to support questions and issues. Although F5 does everything possible to prevent defects in BIG-IP hardware and software, these issues may still arise periodically. Regardless of the root cause of a problem, the goal is to resolve any issues quickly.

F5 technical support offerings

A variety of technical support offerings are available to provide the right level of support for any organization.

F5 Standard and Premium Support include remote assistance from F5 Network Support Engineers, both online and over the phone.

Premium Plus customers receive priority status at F5, with fast, easy access to a dedicated team of senior-level, F5-certified Network Support Engineers and a Technical Account Manager.

To learn more, see **F5 Technical Support Services** or send email to services@f5.com.

Professional services

Take advantage of the full range of F5 Consulting Services to help you design, customize, and implement a solution that is right for your IT infrastructure and which supports your business goals.*

Consulting Services (f5.com/support/services) provides information on a wide range of F5 Professional Services offerings and Professional Services Partners. You can use our online forms to request Consulting Services On Demand for custom, shorter scope consulting engagements, or iRules OnDemand to get fast access to iRules scripts tailored to your specific needs.

You can make an online request for specific support services by filling out a request form:

- ***Consulting request form*** (f5.com/support/professional-services/consulting-request-form).
- ***iRules consulting request form*** (f5.com/support/professional-services/irules-consulting-request-form).

GUARDIAN professional services partners

F5 GUARDIAN Professional Services Partners are authorized as Installation Providers and are also available to assist you. F5 GUARDIANs are selected because they have the skills and experience required to ensure successful implementations of F5 BIG-IP Local Traffic Manager (LTM) and BIG-IP Global Traffic Manager (GTM).

See ***F5 GUARDIAN Professional Service Partners*** (f5.com/support/professional-services#guardian) for a complete list of partners.

F5 certification

F5 Certified exams test the skills and knowledge necessary to be successful when working with today's application delivery challenges. Our technically relevant and appropriate exams deliver consistently reproducible results that guarantee excellence in those that achieve certification.

Certification levels

The F5 certification program is progressive with the four levels – Administrator, Specialist, Expert and Professional -- building on the skills and knowledge demonstrated on previous exams.

C1 – F5 Certified BIG-IP Administrator (F5-CA)

The starting point for all certifications: a certified BIG-IP Administrator has basic network and application knowledge to be successful in application delivery.

C2 – F5 Certified Technology Specialists (F5-CTS)

The Technology Specialist certification assures employers that the candidate is fully qualified to design, implement, and maintain that specific product and its advanced features.

C3 – F5 Certified Solution Expert (F5-CSE)

The Solution Expert focuses on how F5 technologies combine with industry technology to create real-world business solutions.

C4 – F5 Certified Application Delivery Engineer (F5-CADE)

The Application Delivery Engineer certification exam and requirements are still under development.

C5 – F5 Certified Application Delivery Architect (F5-CADA)

The Application Delivery Architect certification exam and requirements are still under development.

Certificate expiration

F5 certifications are valid for two (2) years. Three months before the expiration date, the holder becomes recertification-eligible and can register for the exam necessary to recertify. Only the last exam in the highest level certification achieved needs to be retaken.

Certification beta program

We use Beta exams in the creation of all our exams and to maintain their relevancy and accuracy after production. Beta exams are open to all and give candidates an opportunity to have an impact on the Certified program. While Beta exams are twice as long, they cost less than regular exams and give candidates the chance to leave feedback on the exam. Beta exams are critical to our exam development process and a great way to change the Certified program for the better.

Get involved

There are a several ways to get involved with the F5 certification beta program:

- **Beta participation**. Interested in taking our Beta exams? Contact us at F5Certification@f5.com to learn more.
- **Exam development**. Contact us at F5Certification@f5.com if you're interested in helping us create our Certified! exams.
- **LinkedIn community**. Join us on LinkedIn (this link sends you to a external site) for answers to frequently asked questions, community developed resources, and more.

Visit F5-Credential Management System (certification.f5.com) for information or follow the steps to get registered.

Self help

F5 offers a number of resources to assist in managing and supporting your F5 systems:

- AskF5 Knowledge Base
- Downloads
- Security Updates
- BIG-IP iHealth
- TechNews
- RSS Feeds
- DevCentral
- F5 Global Training Services

AskF5

AskF5 (support.f5.com) is a great resource for thousands of solutions to help you manage your F5 products more effectively and should be the first resource you choose when in need of support. Step-by-step instructions, downloads, and links to additional resources give you the means to solve known issues quickly and without delay, and to address potential issues before they become reality.

Whether you want to search the knowledge base to research an issue, or you need the most recent news on your F5 products, AskF5 is your source for:

- Product manuals, operations guides, and release notes.
- F5 announcements.
- General solutions.
- Known issues.
- Security advisories.
- Recommended practices.
- Troubleshooting tips.
- How-to documents.
- Changes in behavior.
- Diagnostic and firmware upgrades.
- Hotfix information.
- Product lifecycle information.

Downloads

Downloads are available from the F5 website. It is highly recommended that your F5 software is kept up-to-date, including hotfixes, security updates, OPSWAT updates, BIG-IP ASM Signature files, and Geolocation database updates. All software downloads are available from ***F5 Downloads*** (downloads.f5.com).

Security updates

You can receive timely security updates and BIG-IP Application Security Manager (BIG-IP ASM) attack signature updates from F5. When remote vulnerabilities are discovered, F5 implements, tests, and releases security hotfixes for any vulnerable supported version, and sends an email alert to the F5 Security mailing list. F5 encourages customers with an active support account to subscribe to this list. For more information, see AskF5 article: ***SOL4602: Overview of the F5 security vulnerability response policy***.

BIG-IP iHealth

The ***BIG-IP iHealth*** (iHealth.f5.com) diagnostic viewer is among the most important preventative tools to verify the proper operation of your BIG-IP system. It will ensure hardware and software are functioning at peak efficiency and help detect and address issues that may potentially affect F5 systems. BIG-IP iHealth is not integrated within the BIG-IP system. It is hosted by F5 at iHealth.f5.com and can be accessed with any web browser.

F5 recommends you generate a BIG-IP iHealth **qkview** file on the BIG-IP APM system and upload it to iHealth on a weekly basis in order to benefit from the many regularly occuring diagnostic updates. Uploading **qkviews** to iHealth also provides F5 technical support with access to your **qkviews** if you open a support case.

By reviewing the iHealth output, many of the issues commonly experienced by customers can be resolved without the need for opening a support case with F5.

For more information on running BIG-IP iHealth diagnostics, see *BIG-IP iHealth* in ***F5 BIG-IP TMOS: Operations Guide*** or ***BIG-IP iHealth User Guide***.

TechNews

AskF5 provides two TechNews email publications to help keep administrators up-to-date on various F5 updates and other offerings:

- **TechNews Weekly HTML eNewsletter** includes timely information about known issues, product releases, hotfix releases, updated and new solutions, and new feature notices.
- **TechNews Notifications** is a plain-text email that is sent any time a product or hotfix is released. This information is also included in the next weekly HTML TechNews email.

To sign up for the TechNews mailing lists, go to ***AskF5*** (support.f5.com) and select **Subscribe: Mailing Lists** from the **Self-Help** menu. Provide your contact information and select **TechNews Weekly Newsletter** and/or **TechNews Notifications**.

AskF5 recent additions and updates

You can subscribe to F5 RSS feeds to stay informed about new documents pertaining to your installed products or products of interest. *AskF5 Recent Additions and Updates* page provides an overview of all the documents recently added to the Knowledge Base.

Recent Additions and Updates are also published over RSS. You can configure feeds that pertain to specific products, product versions, and/or document sets. You can also aggregate multiple feeds into your RSS Reader to display one unified list of all selected documents.

To generate an RSS feed, go to ***AskF5 Knowledge Base*** and select **Subscribe: RSS** from the **Self-Help** menu.

DevCentral

DevCentral (devcentral.f5.com) is an online forum of F5 employees and customers that provides technical documentation, discussion forums, blogs, media and more, related to application delivery networking. DevCentral is a resource for education and advice on F5 technologies and is especially helpful for iRules and iApps developers. Access to DevCentral is free, but registration is required. As a DevCentral member, you can do the following:

- Ask forum questions.
- Rate and comment on content.
- Contribute to "wikis."
- Download lab projects.
- Join community interest groups.
- Solve problems and search for information.
- Attend online community events.
- View educational videos.

F5 global training services

F5 Global Training Services provides traditional classroom learning opportunities, live-online training, and free, self-paced online courses to help you get the most out of your investment.

In-person courses

F5 courses are available in multiple training facilities across five continents. Each one combines instructor presentations, classroom discussions and interactive labs. The hands-on learning environment helps provide a fast track to accomplishing your goals.

Virtual instructor-led training

Remote on-line courses mirror classroom training. Participants watch the remote instructor's live lecture online, participate in discussions, and perform lab exercises using remote desktop control.

Free online training

You can use the self-paced Getting Started series of free, web-based courses to learn how to deploy F5 solutions to address your most common application delivery problems:

For more information about F5 education opportunities at F5, go to https://f5.com/education

F5 Training Programs and Education (f5.com/education/training) provides links to course schedules, pricing, and registration details. It also has information about alternative training solutions such as virtual and web-based training for those who cannot attend training in person. Links to more information are provided at this site for those interested in F5 Professional Certification or a non-accredited Application Delivery Networking Certificate through F5 and the University of Phoenix.

Engage support

F5 Technical Support is designed to provide support for specific break-fix issues for customers with active support contracts. For more information about F5 scope of support, refer to the ***Support Policies*** article on F5.com.

Options for assistance

You can contact F5 Support in two ways:

- **Online**: You can open a support case at the ***F5 WebSupport Portal***. Click **Register for an Account** to access to the WebSupport Portal.
- **By phone**: Phone numbers are provided in the **General contact numbers** section below. It is strongly recommended that you contact F5 by phone if you have a **Sev1** or **Sev2** case, as defined in the **Opening a support case > Information required when opening a support case** section in this chapter.

F5 technical support resources

F5 support resources are available 24 hours a day, seven days a week, and are distributed around the globe in multiple support centers. Live technical support is provided by our professional Network Support Engineers. Hours of availability may vary depending on the service contract with F5.

Contact numbers

Standard, Premium, and Premium Plus Support customers can open and manage cases by calling one of the contact numbers listed below.

North America

North America: 1-888-882-7535 or (206) 272-6500
Traffix® Support Only: 1-855-849-5673 or (206) 272-5774

Outside North America

Outside North America, Universal Toll-Free: +800 11 ASK 4 F5 or (800 11275 435)

Additional contact numbers by country

Australia: 1800 784 977

China: 010 5923 4123

Egypt: 0800-000-0537

Greece: 00-800-11275435

Hong Kong: 001-800-11275435

India: 000-800-650-1448; 000-800-650-0356 (Bharti Air users)

Indonesia: 001-803-657-904

Israel: 972-37630516

Japan: 81-3-5114-3260 or 0066-33-812670

Malaysia: 1-800-814994

New Zealand: 0800-44-9151

Philippines: 1-800-1-114-2564

Saudi Arabia: 800-844-7835

Singapore: 6411-1800

South Africa: 080-09-88889

South Korea: 002-800-11275435

Taiwan: 00-800-11275435

Thailand: 001-800-12-0666763

United Arab Emirates: 8000-3570-2437

United Kingdom: 44-(0)8707-744-655

Vietnam: 120-11585

Open a support case

F5 provides several resources to help find solutions to problems. Before opening a support case with F5 technical support, check to see if the issue you are encountering is already documented.

The following is a list of resources to consult before opening a support case with F5:

- Deployment guides and white papers provide information about specific deployment configurations.
- *AskF5 Knowledge Base* provides many articles including known issues, how-to guides, security issues, release notes, and general information about products. Many of the issues customers encounter are already documented on this site.
- *BIG-IP iHealth* enables customers to upload **qkview** configuration snapshots in order to verify operation of any BIG-IP system.
- The *Troubleshooting* chapter of this guide provides assistance with remedying some common issues specific to BIG-IP APM.

Gather information to open a support case

If your issue cannot be solved using the resources listed, and you need to open a support case, you must first gather several pieces of important information about your issue. Providing full and accurate information will help speed the path to resolution. The required information for the majority of situations is summarized below:

- **The serial number or base registration key** of the specific BIG-IP system requiring support. For more information, see AskF5 article: ***SOL917:*** *Finding the serial number or registration key of your F5 device*.

- **A full description of the issue**. A clear problem statement is the best tool in helping to troubleshoot issues. Your description should include as much of the following information as you can provide.
    - **Occurrences and changes:** The date and times of initial and subsequent recurrences. Did this issue arise at implementation or later? Were there any changes or updates made to the BIG-IP system prior to the issue arising? If so, what were they?

  - **Symptoms**: Ensuring your list of symptoms is as detailed as possible will give more information for support personnel to correlate with.
  - **Scope of the problem**: Note whether the problem is system-wide or limited to a particular configuration feature, service, or element (such as VLAN, interface, application service, virtual server, pool, and so on).
  - **BIG-IP APM component:** The feature, configuration element, or service being used when the problem occurred (for example: portal access, network access, authentication services, VDI, Exchange).
  - **Steps to reproduce:** The steps to reproduce the problem as accurately and in as much detail as possible. Include expected behavior (what *should* happen) as well as actual behavior (what *does* happen).
  - **Errors:** Complete text of any error messages produced.
  - **Environment:** Current usage of the system. (Is this unit in production? If so, is there currently a workaround in place?)
  - **Browers:** Types and versions, if applicable.
  - **Changes**: System changes made immediately prior to the problem's first occurrence. This may include upgrades, hardware changes, network maintenance, and so on. Have any changes been made to resolve the problem? If so, what were they?

- **Issue Severity:** A description of the impact the issue is having on your site or Case severity
  - *Severity 1*: Software or hardware conditions on your F5 device are preventing the execution of critical business activities. The device will not power up or is not passing traffic.
  - *Severity 2*: Software or hardware conditions on your F5 device are preventing or significantly impairing high-level commerce or business activities.
  - *Severity 3*: Software or hardware conditions on your F5 device are creating degradation of service or functionality in normal business or commerce activities.
  - *Severity 4:* Questions regarding configurations ("how to"), troubleshooting non-critical issues, or requests for product functionality that are not part of the current product feature set.

- **Contact and availability information** including alternate contacts authorized to work on the problem with F5 Technical Support. When there are more personnel available to work with F5 Technical Support, the resolution of your issue may be expedited.

- **Remote access information**, if possible.

- A **qkview** file obtained while problem symptoms are manifesting. A **qkview** of the system before the occurrence is also useful. F5 recommends archiving **qkviews** regularly, refer to the ***F5 TMOS Operations Guide*** for details.

- **Product-specific information**: Software versions and types of equipment in use.

- **Platform and system**. Version and provisioned software modules of the affected system.

To locate platform and system information using tmsh from the command line

- Type the following command:

```
tmsh show /sys hardware
```

Output will appear similar to the following example:

```
<SNIP some of the output>
```

```
Platform
Name  BIG-IP 3900
BIOS Revision  F5 Platform: C106 OBJ-0314-03 BIOS (build: 010) Date: 02/15/12
Base MAC       00:01:d7:be:bf:80
 System Information
Type                  C106
Chassis Serial        f5-jspv-lzxw
Level 200/400 Part    200-0322-02 REV C
Switchboard Serial
   Switchboard Part Revision
Host Board Serial
Host Board Part Revision
```

To copy software version and build number information from the command line

1. Type the following command:

```
cat /VERSION
```

Output will appear similar to the following example:

```
Product: BIG-IP
Version: 11.6.0
Build: 0.0.401
Sequence: 11.6.0.0.0.401.0
BaseBuild: 0.0.401
Edition: Final
Date: Mon Aug 11 21:08:03 PDT 2014
Built: 140811210803
Changelist: 1255500
JobID: 386543
```

2. Highlight and copy the output information and include it with your support case.

To copy provisioned module information from the command line

1. Type the following command:

```
tmsh list /sys provision
```

Output will appear similar to the following example:

```
sys provision afm { }
sys provision am { }
sys provision apm {
level nominal
}
sys provision asm { }
sys provision avr { }
sys provision fps { }
sys provision gtm { }
sys provision lc { }
sys provision ltm {
level minimum
}
sys provision pem { }
sys provision swg { }
```

2. Highlight and copy the output information and include it with your support case.

Open a case using WebSupport Portal

If you cannot find the answer to your problem using the resources listed above, you can open a support case online, using the ***F5 WebSupport Portal*** (websupport.f5.com).

Use of the WebSupport Portal requires a current support contract and registration on the F5 website (login.f5.com).
To request access during registration, select **I have a support contract and need access to WebSupport**. You will be prompted to enter your registration key or serial number. Once registered, you'll receive an email within 24 hours letting you know your account has been enabled with WebSupport Portal access.

To register for WebSupport portal access

1. Go to ***F5 WebSupport portal***.
2. Click **Register for an Account**.
3. Enter your email address.
4. Complete the Contact information portion of the page and then select **I have a support contract and need access to WebSupport**.
5. Enter your Serial Number or Registration Key (optional).

After you have logged-in you are ready to open a support case.

Send information to support

Once the information is assembled and appropriate documentation gathered, transfer it to F5 technical support following the procedures contained in the **Share diagnostic files with F5 technical support** section of this guide. For more information, see AskF5 article: ***SOL2486:*** *Providing files to F5 Technical Support*.

For more information about opening a support , see AskF5 article: ***SOL11898:*** *Information required when opening a support case for BIG-IP APM*.

Collect BIG-IP APM data

To open a support case for BIG-IP APM, additional module-specific data collection may be necesssary to give support engineers a complete understanding of your system's issues. Depending upon the BIG-IP APM access method or feature at issue, F5 support may request this additional information. Use the following section to guide you in collecting the this information.

Change logging levels

You can also use the BIG-IP APM log messages to assist in data collection. To do so, you may need to change the BIG-IP APM log level to debug.

Important Logging at the debug level creates large log files which may lower performance. Once you are finished troubleshooting, you will need to return the logging level (**Notice**).

To enable additional Access Policy logging using the Configuration utility

1. Go to **System** > **Logs** > **Configuration** > **Options** > **Access Policy**.
2. In the **Access Policy Logging** section, for **Access Policy, SSO, Portal Access,** and **VDI,** select **Debug**.
3. Click **Update.**

Tip To return to default logging level, set each item to **Notice**.

To enable additional logging using tmsh at the command line

- Type the following commands:

```
tmsh modify sys db log.accesscontrol.level { value "debug" }
tmsh modify sys db log.sso.level { value "debug" }
tmsh modify sys db log.webapplications.level { value "debug" }
tmsh modify sys db log.vdi.level { value "debug" }
```

To disable additional logging using tmsh at the command line

- Type the following commands:

```
tmsh modify sys db log.accesscontrol.level { value "notice" }
tmsh modify sys db log.sso.level { value "notice" }
tmsh modify sys db log.webapplications.level { value "notice" }
tmsh modify sys db log.vdi.level { value "notice" }
```

Network captures

In most issues involving BIG-IP APM, having network packet trace information available is beneficial. As BIG-IP APM is a full proxy, a trace that captures traffic on both the client side and server side is recommended. Utilities such as **tcpdump** and **ssldump** can capture traffic.

For more information on these utilities, see AskF5 articles: ***SOL13637: Capturing internal TMM information with tcpdump*** and ***SOL10209: Overview of packet tracing with the ssldump utility***.

Notes:

- For versions BIG-IP APM 11.2.0 and higher, **ssldump** utility should be run with the **-M** command line switch to generate the pre-master secret (PMS) key log file. The decrypted output contains the SSL certificate information required to decrypt network captures. This data should be included with your support case information and will be requested by Support if it is not.
- When multiple traffic captures using **tcpdump** utility are required, be sure to run all captures concurrently before sending test traffic.

Tip The **ssldump** utility can only decrypt data if the client and server use RSA for key negotiation. You may need to temporarily force the client and server to use RSA for their SSL session.

For more information, see ***SOL10209 Overview of packet tracing with the ssldump utility***.

Network access data collection

For issues related to network access, data will need to be collected both from the BIG-IP system and on the client system.

BIG-IP APM network capture

A network capture taken on the BIG-IP APM connectivity profile interface will show all VPN traffic as it arrives from the client or traffic received from the local LAN to be sent to a VPN client decrypted:

```
tcpdump -nnvi <connectivity_profile>:nnn -s0 -w /var/tmp/<f5-issue-id-ddmmyy>.pcap
```

To view traffic entering and leaving the BIG-IP via the local VLAN interfaces from the command line

- Type the following command:

```
tcpdump -nnvi 0.0:nnn -s0 -w /var/tmp/<f5-issue-id-ddmmyy>.pcap
```

Tip The **tcpdump** utility captures can get quite large, therefore you should attempt to start the capture immediately before and stop the capture immediately after the behavior you want to send to F5 Support. Optionally, you can add a **-c** command to the **tcpdump** command line to indicate how many packets to capture. Follow the **-c** option with a packet number value. For example:

```
 tcpdump -nnvi 0.0:nnn -s0 -c 5000000 -w /var/tmp/<f5-issue-id-
ddmmyy>.pcap
```

Windows Remote Access Service

F5 VPN components rely on Microsoft Remote Access Service (RAS) components, so enabling RAS logging will provide details if the issues are occurring on a Windows client.

To enable RAS tracing using a Windows command prompt

- Open a Windows command prompt and type the following command

```
netsh ras set tracing * enabled
```

Once the command has completed, a new network access session can be started. Once the issue occurs, run the following command at the Command Prompt to

To flush RAS logs using a Windows command prompt

- Open a Windows command prompt and type the following command

```
netsh ras set tracing * disabled
```

The resulting log files are located in the **\windows\tracing** directory. The **ppp.log** file should be provided to F5 support.

Client-side logging and network captures are also necessary in order to provide F5 support with an end-to-end view of the issue.

Microsoft network monitor (Netmon) or Wireshark can be used to gather network capture on the client machine. NetMon allows for direct captures on PPP interfaces and has the output exported in the tcpdump-compatible pcap format.

To use Netmon to capture network data

1. Find the application and right-click **Run as Administrator**.
2. From the **Interface Adapter** list, select the active network interface (wireless or wired) and the **NDISWANBH** adapter.
3. Select **New Capture**.
4. Press **Play**.
5. Start the F5 Edge Client.
6. Go to **Details** and select **Enhanced Logging**.
7. Click **Connect** and log in to BIG-IP APM.
8. Once the APM session is established, note the client IP address. If possible, also determine which session ID the user received.
9. Once the issue occurs, disconnect from the Edge Client.
10. In Netmon, click **Stop**.
11. Save the network monitor capture using **File** > **Save As.**

Tip: Session ID information can be found using the Configuration utility in the Access Policy >> Manage Sessions menu.

The traffic captured on **NDISWANBH** will not be encrypted by the TLS/DTLS layer of the F5 VPN Adapter. The F5 **f5wininfo.exe** utility also gathers F5 specific log files from the F5 VPN components installed on the system.

To generate the information needed for support with ftwininfo.exe

1. Download and run the **client troubleshooting** utility. For more information, see Ask F5 article: ***SOL12444:*** *Overview of the Component Troubleshooting Utility*.
2. Click **File.**
3. Click **Generate Report.**
4. Click the **Text** option.
5. Click **Save As**.
6. Enter a path and filename where you want to save the report.
7. Click **Save**.
8. When the report completes, click **Cancel**.

Apple OS X and Linux

For information on how to enable debug logs for the OS X Edge Client, see AskF5 article: ***SOL12321: Enabling Network Access debugging for Mac OS X and Linux***.

Once debugging is enabled, a network capture can be started, and the user can start a network access session. To capture network traffic, Wireshark or **tcpdump** can be used to capture traffic on **tun device tun0**. This device is created only after the network access tunnel is established (i.e. the user is logged on to BIG-IP APM). So this capture would be started in a separate terminal window.

To capture the whole session (such as login, during session, and log out), a simultaneous **tcpdump** dump capture should be run from a terminal on the interface connected to the Internet:

For example:

```
sudo tcpdump -w /path/to/file -nni <interface> -vv -s0 host <Virtual server IP>
```

Portal access data collection

For all portal access issues, the following general data can be collected and provided to F5 Support:

- The access portal resource name experiencing the issue.
- HTTP trace from client direct to web application.
- HTTP trace through BIG-IP APM portal access resource.

Note: Clear the client's browser cookies and cache, then close and reopen the browser before capturing a new HTTP trace.

A number of browser plugins and tools exist for capturing HTTP traces like HTTPWatch, Fiddler, and the developer tools for Google Chrome, Mozilla Firefox and OS X Safari browsers. Preferred file formats are HTTPWatch's HWL file or an HTTP Archive (HAR) file.

HTTP trace from client direct to web application

HTTP traces taken from the client directly to the backend application without BIG-IP APM involved provide a picture of what a successful request or HTTP session should look like. When capturing the HTTP trace, the client must use the server hostname (http://example.com/) and not IP address (http://192.168.1.100/). The HTTP trace **must be started before** the first request.

Note: Clear client browser cookies and cache, then close and reopen the browser before capturing a new HTTP trace.

HTTP trace through BIG-IP APM portal access resource

The HTTP trace taken while accessing the portal access resource on the BIG-IP APM will show the application failure. BIG-IP APM provides an additional tool for gather diagnostics data via the Web Application Trace functionality. For more information, see AskF5 article: ***SOL13384: Performing a web applications trace (11.x)***.

Note: Clear the client's browser cookies and cache, then close and reopen the browser before capturing a new HTTP trace. Also, start the capture at the point of launching the portal access favorite, and end the capture after error condition.

JavaScript errors

If the application is throwing JavaScript errors, use the following steps to enable and view the JavaScript errors in the following web browsers:

- Internet Explorer : **Tools > Internet Options > Advanced > Display a notification about every script error**.
- Firefox: **Menu > Developer > Web Console**.
- Chrome: **Menu > Tools > Javascript Console**.

Once all data is gathered, return the access policy and portal access log settings to their defaults.

Java rewrite

If a portal access issue deals with a Java Applet, make sure of the following:

- A **Web Acceleration** profile is applied on the BIG-IP APM virtual server and the URL matching the jar file paths in the **URI List**.
- The portal access resource has the Java Patching enabled.
- The portal access resource item has a correctly matching path defined which matches the jar file paths .

If the issue still persists after validating these steps, do the following:

- Gather the Java console logs from the client machine as described in *Java Console, Tracing, and Logging* in ***Java Platform, Standard Edition Deployment Guide***. (This link sends you to an external site.)
- Make sure that the Java console debug logs when accessing the application directly, bypassing BIG-IP APM.
- Take a network capture taken on the client machine.
- Take a network capture on the BIG-IP system Internal VLAN connected to the backend Web Application server.

To take a network capture of the Internal VLAN connected to the backend Web Application server from the command line

- Type the following command:

```
  tcpdump -nnvi <internal_vlan_name>:nnn -s0 -w /var/tmp/<f5-issue-id-ddmmyy>.pcap host
<web_application_server_ip>
```

Application access data collection

Application tunnels are split tunnels. Before troubleshooting, make sure of the following on the client machine:

- DNS Relay Proxy service is up and running.
- Component Installer service up and running.
- User has administrative rights.

If you cannot verify any of these conditions, make sure that the DNS relay proxy and component installer services are installed. These components are required to allow application tunnels to run correctly without user elevation privilege or name resolution issues.

ActiveX and RDP application tunnels

The **f5wininfo.exe utility** also gathers F5 specific log files from the F5 application tunnel components installed on the system.

To generate the information needed for support using the f5wininf.exe utility

1. Download and run the **client troubleshooting** utility.
2. Click **File.**
3. Click **Generate Report.**
4. Click the **Text** option.
5. Click **Save As**.
6. Enter a path and filename where you want to save the report.

7. Click **Save**.
8. When the report completes, click **Cancel**.

For more information, see AskF5 article: ***SOL12444**: Overview of the Component Troubleshooting Utility*.

Tip: Running the **f5wininfo.exe** utility on a user system that is not experiencing the same issue is helpful.

Java application tunnels

Java Application Tunnels can be run on Windows, Macintosh, or Linux clients. The following troubleshooting steps will work on any of these operating systems.

- Gather the Java console logs from the client machine as described in *Java Console, Tracing, and Logging* in ***Java Platform, Standard Edition Deployment Guide***. (This link takes you to an external site.)
- Make sure that Java console debug logs when accessing the application directly, bypassing BIG-IP APM.
- Take a network capture taken on the client machine.
- Take a network capture on the BIG-IP system Internal VLAN connected to the backend Web Application server.

To take a network capture of the Internal VLAN connected to the backend Web Application server from the command line

- Type the following command:

```
tcpdump -nnvi <internal_vlan_name>:nnn -s0 -w /var/tmp/<f5-issue-id-ddmmyy>.pcap host
<web_application_server_ip>
```

Per-app VPN data collection

When experiencing issues with a Per-app VPN, the following information needs to be collected:

- Mobile OS type (such as iOS or Android).
- BIG-IP EdgeClient version being used.
- BIG-IP EdgeClient logs.

- Per-app VPN **mobileconfig** file (if used to provision mobile client).
- Network trace on 0.0 interface while reproducing issue:

```
tcpdump -nnvi 0.0:nnn -s0 -w /var/tmp/<f5-issue-id-ddmmyy>.pcap
```

VDI Issues data collection

BIG-IP APM provides services for:

- Citrix XenDesktop/XenApp
- VMWare View
- RDP

Citrix

When collecting data for Citrix issues, provide the following information:

- The BIG-IP APM deployment in use? (such as Web Interface, storefront, or portal mode).
- Whether a Citrix XenApp server is in use and if so, the version.
- Whether a Citrix XenDesktop server is in use and if so, the version.
- Whether a Citrix StoreFront server is in use and if so, the version.
- Whether a Citrix Web Interface server is in use and if so, the version.
- Whether a Citrix Receiver is in use.
    - Operating system in use: (such as iOS, Android, Windows).
    - Citrix Receiver client version in use.

In addition to a **qkview**, combine the BIG-IP APM logs by typing the following command:

```
tail -f /var/log/{apm,ltm} > /var/tmp/f5-issue-id-ddmmyy-citrix.log
```

In a separate terminal window, type the following command:

```
tcpdump -s0 -nni 0.0:nnn (host <client_ip> or host <xml_broker(s)_ip(s)> or host
<connection_resource_ip>) -w /var/tmp/f5-issue-id-ddmmyy-citrix.pcap
```

VMWare View

When collecting data for VMware issues, provide the following information:

- Whether a VMware Horizon View in use and if so, the version.

- Whether a Connection server in use and if so, the version.
- Whether a vCenter server in use and if so, which version.
- Whether a VMware Horizon View Client in use and if so, the version.
- The operating system in use and its version.
- The HTML5 connection in use if applicable.
- Whether the desktop resource have HTML Access software installed on it.

In addition to a **qkview**, combine the BIG-IP APM logs with the following command:

```
tail -f /var/log/{apm,ltm} > /var/tmp/f5-issue-id-ddmmyy-view.log
```

In a separate terminal window, run the following command:

```
tcpdump -s0 -nni 0.0:nnn (host <client_ip> or host <view_connection_server_ip(s)>
or host <view_security_server_ip(s)> or host <vdi_desktop_server_ip(s)) -w
/var/tmp/f5-issue-id-ddmmyy-view.pcap
```

Authentication and single sign-on data collection

Authentication issues with AAA resources occur usually between the BIG-IP APM and AAA resource on an internal VLAN. A network capture between the BIG-IP APM and AAA resource will provide the most insight in most cases:

```
tcpdump -nnvi <internal_vlan_name>:nnn -s0 -w /var/tmp/<f5-issue-id-ddmmyy>.pcap
```

In addition, the following information should be collected:

- Session report for the failed user session.
- BIG-IP APM logs files in debug level (provided in qkview).
- Name of the AAA resource being used.
- Network capture taken on the user's system if Kerberos authentication is being used to authenticate users.

SAML

BIG-IP APM can be deployed as an IdP or SP to provide SAML federation to clients. When troubleshooting issues for federated access, the following data should be gathered:

- Use Case: IdP or SP initiated.

- HTTPWatch HWL, HAR archive or SAML Trace log (Firefox only) session.
- XML before and after canonicalization from application.
- Metadata from the partner device.
- A test account on the partner device to the BIG-IP APM (IdP or SP) and associated metadata.
- A network capture on interface which sees the SAML traffic.

Single sign-on

Single sign-on (SSO) issues occur between the BIG-IP APM system and a backend application. A network capture between the BIG-IP APM and the back-end application will provide the most insight in most cases:

```
tcpdump -nnvi <internal_vlan_name>:nnn -s0 -w /var/tmp/<f5-issue-id-ddmmyy>.pcap
```

In addition, the following information must be provided:

- Name of the SSO resource in use.
- SSO Type (e.g. Basic, Forms, Kerberos, etc).
- What backend application uses for authentication (e.g. IIS w/ Forms or Basic (etc.), Apache w/ mod_xx.., Oracle Access Manager, etc.)
- An **mpidump** capture - see ***SOL11765****: Frequently used tools for troubleshooting BIG-IP APM and Edge Gateway issues (10.x)* for more details

OPSWAT (antivirus/firewall) data collection

In cases where a client endpoint inspection check fails or does not work correctly, the following information must be collected:

- Session report for the failed user session.
- Download, install and run the latest OPSWAT Security Score tool (this link goes to an external site).
- Steps needed for Windows and OS X.

High-resource utilization

BIG-IP APM uses multiple system daemons to provide its functionality. There may be times when certain processes start using too much CPU time or in a hung state. These are described in Ask F5 article ***SOL15263: BIG-IP APM daemons (11.x)***. When one of these daemons are suspected of causing load-related issues with the BIG-IP APM system, the following commands should be run from the BIG-IP command line:

```
top -cbn 5 > /var/tmp/<f5-issue-id-ddmmyy>-top.txt
qkview -s0 -C
```

The **qkview** file can then be provided to F5 support and will contain the output from the previous top command.

Share diagnostic files with F5 technical support

F5 technical support may require diagnostic files to help resolve technical support issues. Upload files to F5 using one of the following two methods:

- Upload **qkview** diagnostic files to *BIG-IP iHealth* (ihealth.f5.com).
- Upload/downloading files using dropbox.f5.com.

Upload qkview diagnostic files to BIG-IP iHealth

The preferred method for providing a **qkview** diagnostic file to F5 Support is to upload the file to the ***BIG-IP iHealth* website**. BIG-IP iHealth allows you to quickly diagnose the health and proper operation of your BIG-IP system. For more information about using BIG-IP iHealth, see the **BIG-IP iHealth** chapter in the ***TMOS Operations Guide***.

Upload/download files using dropbox.f5.com

The dropbox.f5.com site is a widely available file repository for exchanging incoming and outgoing diagnostic files with the F5 Technical Support team. The dropbox.f5.com site supports HTTP, FTP, and SFTP for transferring files to F5, and FTP and SFTP for retrieving files from F5.

Username and password

Access to the dropbox.f5.com site is associated with an open support ticket number with syntax CXXXXXX or 1-########. The username provided to the dropbox.f5.com site is the ticket number, and the password provided is an email address of a user associated with the ticket.
For example, if joeuser@example.com has opened ticket C123456, he would log in to the dropbox.f5.com site using the following information:

```
Username: C123456
Password: joeuser@example.com
```

If joeuser@example.com has opened ticket 1-12345678, he would log in to the dropbox.f5.com site using the following information:

```
 Username: 1-12345678 Password: joeuser@example.com
```

For additional information regarding uploading and downloading files using dropbox.f5.com, please refer to ***SOL2486: Providing files to F5 Technical Support***.