

# Configuration Guide for BIG-IP® Access Policy Manager®

version 11.4

MAN-0309-05





---

## Product Version

This manual applies to product version 11.4 of the BIG-IP® Access Policy Manager® product.

## Publication Date

This manual was published on September 27, 2013.

## Legal Notices

### Copyright

Copyright 2007-2013, F5 Networks, Inc. All rights reserved.

F5 Networks, Inc. (F5) believes the information it furnishes to be accurate and reliable. However, F5 assumes no responsibility for the use of this information, nor any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent, copyright, or other intellectual property right of F5 except as specifically described by applicable user licenses. F5 reserves the right to change specifications at any time without notice.

### Trademarks

AAM, Access Policy Manager, Advanced Client Authentication, Advanced Firewall Manager, Advanced Routing, AFM, Alive With F5, 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, Edge Portal, ELEVATE, EM, Enterprise Manager, ENGAGE, F5, F5 [DESIGN], F5 Certified [DESIGN], F5 Networks, 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 Rate Shaping, LC, Link Controller, Local Traffic Manager, LTM, LineRate, LineRate Systems [DESIGN], LROS, Message Security Manager, MSM, OneConnect, Packet Velocity, PEM, Policy Enforcement Manager, Protocol Security Manager, PSM, Real Traffic Policy Builder, ScaleN, Signalling Delivery Controller, SDC, SSL Acceleration, StrongBox, SuperVIP, SYN Check, TCP Express, TDR, TMOS, Traffic Management Operating System, Trafix Systems, Trafix Systems (DESIGN), Transparent Data Reduction, UNITY, VAULT, VIPRION, vCMP, VE F5 [DESIGN], Virtual Clustered Multiprocessing, WA, WAN Optimization Manager, WebAccelerator, WOM, and ZoneRunner, are trademarks or service marks of F5 Networks, Inc., in the U.S. and other countries, and may not be used without F5's express written consent.

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

### Patents

This product protected by U.S. Patents 6,505,230, 7,114,180, and 7,349,391. Other patents may be pending.

### Export Regulation Notice

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

### RF Interference Warning

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

### FCC Compliance

This equipment has been tested and found to comply with the limits for a Class A digital device pursuant to Part 15 of FCC rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This unit generates, uses, and

---

can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case the user, at his own expense, will be required to take whatever measures may be required to correct the interference.

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

## Canadian Regulatory Compliance

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

## Standards Compliance

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

## Acknowledgments

This product includes software developed by Bill Paul.

This product includes software developed by Jonathan Stone.

This product includes software developed by Manuel Bouyer.

This product includes software developed by Paul Richards.

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

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

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

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

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

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

This product includes software developed by Adam Glass.

This product includes software developed by Christian E. Hopps.

This product includes software developed by Dean Huxley.

This product includes software developed by John Kohl.

This product includes software developed by Paul Kranenburg.

This product includes software developed by Terrence R. Lambert.

This product includes software developed by Philip A. Nelson.

This product includes software developed by Herb Peyerl.

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

This product includes software developed by Chris Provenzano.

This product includes software developed by Theo de Raadt.

This product includes software developed by David Muir Sharnoff.

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

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

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

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

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

This product includes software developed by Christos Zoulas.

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

In the following statement, "This software" refers to the Mitsumi CD-ROM driver: This software was developed by Holger Veit and Brian Moore for use with "386BSD" and similar operating systems.

"Similar operating systems" includes mainly non-profit oriented systems for research and education, including but not restricted to "NetBSD," "FreeBSD," "Mach" (by CMU).

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

---

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

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

This product includes software developed by Jared Minch.

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

This product includes cryptographic software written by Eric Young ([ey@cryptsoft.com](mailto:ey@cryptsoft.com)).

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

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

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

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

This product includes Hypersonic SQL.

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

This product includes software developed by the Internet Software Consortium.

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

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

---



---

---

# Table of Contents

---

---





# I

## Introducing BIG-IP Access Policy Manager

Introducing the BIG-IP system .....	I-1
BIG-IP Local Traffic Manager .....	I-1
Overview of the BIG-IP Access Policy Manager .....	I-2
Introducing Access Policy Manager features .....	I-2
Understanding BIG-IP Access Policy Manager access types .....	I-4
Working with network access .....	I-6
Working with portal access .....	I-9
Working with application access .....	I-11
Working with web access management .....	I-13
Using access profiles and policies .....	I-16
Using authentication in access policies .....	I-17
Using the Configuration utility .....	I-19
Overview of components of the Configuration utility .....	I-20
Getting started with BIG-IP Access Policy Manager .....	I-21
Using Access Policy Manager configuration wizards .....	I-21
Following the recommended configuration path .....	I-25
Possible configuration scenarios .....	I-26
Finding help and technical support resources .....	I-27
Finding the Access Policy Manager software version number .....	I-27

# 2

## Configuring Web Access Management

Introducing web access management .....	2-1
Understanding how web access management works .....	2-1
Reviewing web access management options .....	2-2
Setting timeouts for web access management policy management .....	2-2
Understanding other web access management considerations .....	2-3
Configuring web access management .....	2-4

# 3

## Configuring Resources

Understanding resources .....	3-1
Using access control lists .....	3-2
Creating static access control lists .....	3-2
Access control list examples .....	3-6
Configuring dynamic ACLs .....	3-8
Understanding dynamic ACLs .....	3-8
Understanding the F5 ACL format .....	3-8
Understanding the Cisco ACL format .....	3-10
Creating a dynamic ACL container .....	3-11
Adding a dynamic ACL to an access policy .....	3-12
Using webtops .....	3-14
Using AD query with IPv6 .....	3-16

# 4

## Understanding Access Policies

Introducing access policies .....	4-1
Understanding access policy items .....	4-2
Understanding the access policy start point .....	4-2
Understanding access policy actions .....	4-2
Understanding authentication actions .....	4-7

Understanding access policy branch rules .....	4-8
Viewing rules .....	4-9
Predefined rules .....	4-10
Understanding access policy branches .....	4-12
About swapping access policy branches .....	4-13
Understanding access policy macros .....	4-14
Introducing macro terminals .....	4-15
Introducing access policy endings .....	4-17
Understanding the allow ending .....	4-17
Understanding the deny ending .....	4-17
Understanding the redirect ending .....	4-18
Understanding session variables .....	4-19
Using session variables .....	4-20

## 5

### Creating Access Profiles and Access Policies

Creating an access profile .....	5-1
Understanding access profile settings .....	5-1
Understanding configuration settings .....	5-2
Understanding Single-Sign On settings .....	5-2
Creating an access profile .....	5-4
Applying an access policy .....	5-4
Customizing access profile languages .....	5-5
Creating an access policy .....	5-7
Starting the visual policy editor .....	5-7
Configuring a basic access policy .....	5-8
Opening an access policy .....	5-9
Adding actions to an access policy .....	5-9
Using policy endings .....	5-10
Applying an access policy configuration .....	5-14
Configuring macros .....	5-15
Using predefined macro templates .....	5-18
Using the empty macro template .....	5-18
Using the AD auth and resources macro template .....	5-18
Using the SecurID and resources macro template .....	5-19
Exporting and importing access profiles .....	5-21

## 6

### Configuring Logon, Assignment, and General Purpose Actions

Configuring actions in an access policy .....	6-1
Adding and customizing a logon page .....	6-1
Adding an HTTP 401 response page .....	6-5
Adding an external logon page .....	6-7
Assigning resources .....	6-8
Assigning variables .....	6-10
Adding a virtual keyboard to the logon screen .....	6-13
Adding SSO credential mapping .....	6-14
Filtering access with Citrix SmartAccess filters .....	6-15
Selecting a route domain or SNAT .....	6-16
Adding access policy logging .....	6-17
Adding a message box .....	6-17
Adding a decision box .....	6-18
Adding a dynamic ACL .....	6-19
Adding an iRule event .....	6-20

## 7

## Configuring Endpoint Security (Client-Side)

Understanding endpoint security (client-side) checks .....	7-1
Checking for a file .....	7-2
Checking for a file with the file check access policy item .....	7-2
Example: Using file check .....	7-4
Checking a machine certificate .....	7-6
About checking a machine certificate on a Windows client .....	7-6
About checking a machine certificate on a Mac client .....	7-6
Understanding machine cert auth check options .....	7-6
Checking a machine certificate with the machine cert access policy item .....	7-9
Example: Using machine cert auth check .....	7-10
Verifying Windows information .....	7-11
Setting up Windows info action .....	7-11
Example: Using Windows info check .....	7-12
Checking machine information .....	7-14
Example: Using machine info check .....	7-16
Checking processes .....	7-19
Setting up the process check access policy item .....	7-19
Example: Using process check .....	7-19
Setting up Windows registry check .....	7-21
Expression syntax .....	7-21
Example: Using the Windows registry action .....	7-22
Setting up Windows cache and session control .....	7-24
Setting up the cache and session control access policy item .....	7-24
Example: Using cache and session control .....	7-26
Setting up Windows protected workspace .....	7-28
Setting up the protected workspace access policy item .....	7-28
Example: Using protected workspace .....	7-30
Assigning a Windows group policy template .....	7-32
Understanding Windows group policy templates .....	7-32
Using predefined Windows group policy templates .....	7-32
Understanding the regulatory templates .....	7-35
Working with Windows group policy templates .....	7-36
Setting up the Windows group policy access policy item .....	7-37
Example: Using Windows group policy templates .....	7-38
Checking software on an endpoint .....	7-40
About supported vendor and product ID lists in software checks .....	7-40
About recurring endpoint checks .....	7-40

## 8

## Configuring Endpoint Security (Server-Side)

Introducing endpoint security (server-side) checks .....	8-1
Preparing for clients that cannot use client checks .....	8-1
Checking the landing URI of a client .....	8-1
Configuring client OS check .....	8-2
Setting up the client OS check .....	8-2
Example: Using client OS check .....	8-3
Configuring cliweb site at ent type check .....	8-5
Setting up the client type access policy item .....	8-5
Example: Using client type check .....	8-6
Checking for client-side check capability .....	8-8
Setting up the client-side check capability access policy item .....	8-8
Example: Using client-side check capability action .....	8-9

Checking a landing URI with the landing URI check .....	8-11
Setting up the landing URI access policy item .....	8-11
Example: Using landing URI check .....	8-11
Identifying Microsoft Exchange clients with the client for MS Exchange check .....	8-14
Understanding Microsoft Exchange connections .....	8-14
Setting up the MS Exchange check policy item .....	8-15
Example: Using client for MS Exchange check .....	8-15
Using IP Geolocation in an access policy .....	8-17
Setting up the IP geolocation match access policy item .....	8-17
Example: Using IP geolocation .....	8-18

## 9

### Using Certificate Authentication in APM

Controlling SSL traffic .....	9-1
Understanding SSL profiles .....	9-1
Introducing SSL server certificates .....	9-2
Understanding APM certificate authentication agents .....	9-3
Client certificate inspection agent .....	9-3
On-Demand certificate authentication agent .....	9-4
Configuring client SSL profiles .....	9-8
Importing a certificate and the corresponding key .....	9-8
Configuring a clientssl profile .....	9-8
Using certificates to authenticate users .....	9-10
Understanding certificate revocation status .....	9-11
Understanding CRLs .....	9-11
Understanding OCSP .....	9-12
Configuring an OCSP responder object .....	9-13
Creating an SSL OCSP profile .....	9-14
Using CRLDP .....	9-15
Configuring a CRLDP server object .....	9-15
Configuring a CRLDP configuration object .....	9-15
Creating a CRLDP profile .....	9-16

## 10

### Configuring Virtual Servers

Introducing virtual servers with Access Policy Manager .....	10-1
Understanding SNAT interactions .....	10-1
Configuring virtual servers for access policies .....	10-2
Creating a virtual server for DTLS .....	10-3
Configuring a local traffic virtual server with an access policy .....	10-5

## 11

### Advanced Topics in Access Policies

Setting up a logon page to collect user credentials .....	11-1
Understanding the logon page action .....	11-1
Example: Using a customized logon page to collect user credentials .....	11-4
Using multiple authentication methods .....	11-7
Client certificate two-factor authentication .....	11-7
Example: Using client certificate authentication with Active Directory .....	11-8
Configuring the client certificate two factor authentication with Active Directory example .....	11-8
Configuring policy routing .....	11-10
Setting up route domain selection in an access policy .....	11-10

Example: Directing users to different route domains .....	11-12
Configuring the policy routing example .....	11-12
Using advanced access policy rules .....	11-16
Understanding advanced access policy rule situations .....	11-16
Writing advanced access policy rules .....	11-17
Using a Tcl expression or program as an advanced access policy rule .....	11-17
Understanding advanced access policy rule limitations .....	11-18
Editing advanced access policy rules .....	11-18
Example: Using a certificate field for logon name .....	11-22
Writing the example code .....	11-22
Using this example .....	11-22

## 12

### Logging and Reporting

Understanding logging .....	12-1
Introducing logging features .....	12-1
Understanding log content .....	12-2
Modifying settings for the log database .....	12-3
Modifying settings for the log file .....	12-3
Understanding log types .....	12-5
Logging system events .....	12-5
Auditing configuration changes .....	12-5
Setting log levels .....	12-7
Setting log levels for auditing events .....	12-8
Understanding reports .....	12-9
Setting the default report .....	12-9
Displaying the All Sessions report .....	12-9
Displaying session variables for current sessions .....	12-10
Configuring and Running Custom Reports .....	12-10
Monitoring system and user information .....	12-13
Viewing the Access Policy Manager dashboard .....	12-13

## 13

### Configuring SNMP

Introducing SNMP administration .....	13-1
Reviewing an industry-standard SNMP implementation .....	13-1
Reviewing the Access Policy Manager system SNMP implementation .....	13-1
Summarizing SNMP configuration on the Access Policy Manager system .....	13-2
Configuring the SNMP agent .....	13-3
Configuring client access .....	13-3
Controlling access to SNMP data .....	13-5
Configuring traps .....	13-7
Working with SNMP MIB files .....	13-10
Downloading SNMP MIB files .....	13-10
Understanding the enterprise MIB files .....	13-11
Collecting performance data .....	13-15
Collecting data on memory use .....	13-16
Collecting data on active connections .....	13-16
Collecting data on new connections .....	13-17
Collecting data on throughput .....	13-18
Collecting data on HTTP requests .....	13-18
Collecting data on RAM Cache utilization .....	13-19
Collecting data on CPU use .....	13-19
Collecting data on active sessions .....	13-21

Collecting data on SSL transactions per second .....	13-21
Additional commands used for SNMP .....	13-22

## 14

### Session Variables

Introducing session variables .....	14-1
Introducing Tcl .....	14-2
Standard operators .....	14-2
Session variables reference .....	14-4
Special purpose user session variables .....	14-11
Understanding network access resource variable attributes .....	14-12
Using session variables in the Configuration utility .....	14-17
Supported fields for session variables in the Configuration utility .....	14-17

## 15

### Using Access iRule Events

Introducing iRules .....	15-1
What is an iRule? .....	15-1
Basic iRule elements .....	15-2
Understanding ACCESS iRules .....	15-4
ACCESS_SESSION_STARTED .....	15-4
ACCESS_POLICY_COMPLETED .....	15-5
ACCESS_ACL_ALLOWED .....	15-5
ACCESS_ACL_DENIED .....	15-6
Using ACCESS_ACL_DENIED .....	15-6
ACCESS_SESSION_CLOSED .....	15-6
ACCESS_POLICY_AGENT_EVENT .....	15-6
Understanding ACCESS iRule Commands .....	15-7
ACCESS::disable .....	15-7
ACCESS::session commands .....	15-7
ACCESS::policy commands .....	15-8

### Glossary

### Index



I

---

# Introducing BIG-IP Access Policy Manager

---

- Introducing the BIG-IP system
- Overview of the BIG-IP Access Policy Manager
- Understanding BIG-IP Access Policy Manager access types
- Using access profiles and policies
- Using the Configuration utility
- Getting started with BIG-IP Access Policy Manager
- Finding help and technical support resources





## Introducing the BIG-IP system

The BIG-IP® system is a port-based, multilayer switch that supports virtual local area network (VLAN) technology. Because hosts within a VLAN can communicate at the data-link layer (Layer 2), a BIG-IP system reduces the need for routers and IP routing on the network. This in turn reduces equipment costs and boosts overall network performance. At the same time, the BIG-IP system's multilayer capabilities enable the system to process traffic at other OSI layers. The BIG-IP system can perform IP routing at Layer 3, as well as manage TCP, UDP, and other application traffic at Layers 4 through 7. The following modules provide comprehensive traffic management and security for many traffic types. The modules are fully integrated to provide efficient solutions to meet any network, traffic management, and security needs.

## BIG-IP Local Traffic Manager

BIG-IP® Local Traffic Manager™ includes features that help make the most of network resources. Using the powerful Configuration utility, you can customize the way that the BIG-IP system processes specific types of protocol and application traffic. By using features such as virtual servers, pools, and profiles, you ensure that traffic passing through the BIG-IP system is processed quickly and efficiently, while meeting all of your security needs. For more information, see ***BIG-IP® Local Traffic Manager™: Concepts*** on <http://www.f5.com>.

## Overview of the BIG-IP Access Policy Manager

The F5® Networks BIG-IP® Access Policy Manager® is a software component of the BIG-IP hardware platform that provides your users with secured connections to Local Traffic Manager virtual servers, specific web applications, or the entire corporate network. By leveraging standard web browsers and security technology, the Access Policy Manager enables your corporation or organization to provide users access to various internal resources easily and cost-effectively, with no special software or configuration on the user's system.

### Introducing Access Policy Manager features

All Access Policy Manager® models include the following features:

- ◆ **Standard Web browser support**  
Access Policy Managers can be used with most standard browsers supporting secure HTTP (also known as HTTPS). These include Internet Explorer, Safari, and Firefox.
- ◆ **Privacy**  
The Access Policy Manager supports common encryption technologies, including RC4, Triple DES, and AES. It uses standard SSL encryption from the client browser to the Access Policy Manager.
- ◆ **Authentication**  
The Access Policy Manager can perform authentication, authorization, and accounting (AAA), using standard AAA methods, including LDAP directories, Microsoft Active Directory and Microsoft Windows Domain servers, RADIUS servers, and HTTP authentication. The Access Policy Manager supports native RSA SecurID authentication. In addition, the controller can use signed client digital certificates to authenticate devices.
- ◆ **Client-side checks**  
The Access Policy Manager provides a broad set of client-side checks such as client integrity checking, browser cache cleaner, secure virtual keyboard, and support for a large number of antivirus and firewall packages.
- ◆ **Visual policy editor**  
To facilitate access policy definition, the Access Policy Manager provides a built-in policy editor that is graphically based, which eases management and supports a visual audit of security access policies.
- ◆ **Administration**  
The Access Policy Manager provides a web-based Configuration utility. The Configuration utility includes tools for managing the Access Policy Manager, configuring secure access, creating and assigning resources, certificate generation and installation, and customization of the remote client user interface.

- ◆ **Web access management**

With Access Policy Manager, you can configure authentication and access control for a web application behind a local traffic virtual server. Using web access management, you create an access policy for a new or existing local traffic virtual server to provide authentication, access control, and endpoint security for the web application.

- ◆ **Network access**

With Access Policy Manager, you can configure a network access VPN connection for remote access. Using network access, you create an access policy and local traffic virtual server so end users can establish a full VPN connection to internal network resources.

- ◆ **Portal access**

With the Access Policy Manager you can configure a remote access connection to one or more internal web applications. Using portal access, you create an access policy and local traffic virtual server so end users can access internal web applications through a single external virtual server. Use this if you need to provide secure extranet access to internal web applications without creating a full VPN connection.

- ◆ **Audit trail**

The Access Policy Manager provides audit tools including full-session audit trails, drill-down session queries, and customizable reports and queries.

- ◆ **High availability**

You can configure Access Policy Managers to fail over to standby controllers, ensuring availability for users.

- ◆ **Scalability**

Access Policy Manager integrates with BIG-IP® system to support large-scale, high-performance deployments, providing universal, secure access for remote, wireless, and internal network users.

- ◆ **BIG-IP system module**

The Access Policy Manager runs as a module of the BIG-IP system. This integration provides a uniform framework that enables users to leverage access policy features with other BIG-IP modules, such as Application Security Manager™.

- ◆ **Client support**

The Access Policy Manager includes web client support for many different systems, including Macintosh and Linux.

- ◆ **BIG-IP Edge Client**

Access Policy Manager is compatible with the BIG-IP Edge Client®, a standalone secure client with robust connection features.

## Understanding BIG-IP Access Policy Manager access types

Access Policy Manager® can be configured to provide four types of access:

- network access
- portal access
- application access (including app tunnels and remote desktops)
- web access management

You use each type of access for a different system scenario. Access Policy Manager provides a set of objects that you can define to provide access to your users through different access methods. You configure Access Policy Manager connections differently for each access type. On the next page, Figure 1.1 shows the configuration of an Access Policy Manager access type. Each access type has common elements and differences. The following table lists the configuration elements that you use to configure each access policy type.

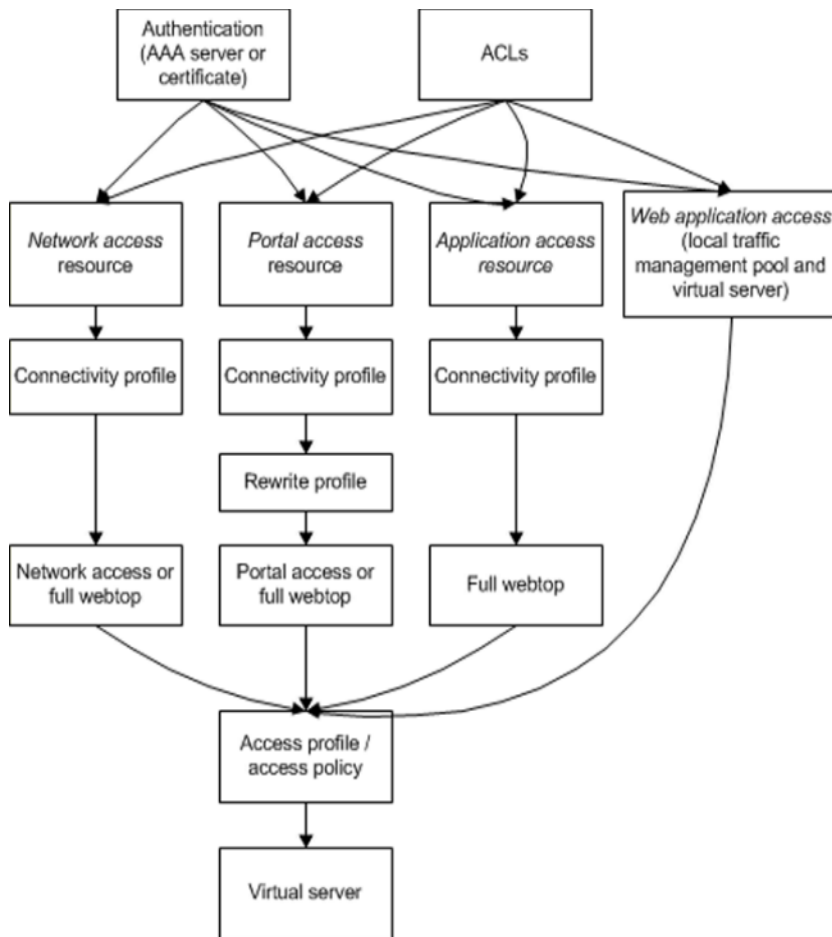
Configuration item	Network access	Portal access	Application access	Web access management
<b>Virtual server</b>	Can use one virtual server for network access, portal access, and application access	Can use one virtual server for network access, portal access, and application access	Can use one virtual server for network access, portal access, and application access	Can use existing local traffic manager virtual server, or create a specific one with the wizard
<b>Local traffic pool</b>	No	Yes	No	Yes, required with at least one member
<b>Access profile and access policy</b>	Yes	Yes	Yes	Yes
<b>Connectivity profile</b>	Yes	Yes	No	No
<b>Rewrite profile</b>	No	Yes	No	No
<b>Network access resource</b>	Yes	No	No	No
<b>Portal access resource</b>	No	Yes	No	No
<b>App tunnel resource</b>	No	No	Yes	No
<b>Remote desktop resource</b>	No	No	Yes	No

**Table 1.1** Configuration elements for Access Policy Manager access types

Configuration item	Network access	Portal access	Application access	Web access management
<b>Authentication</b>	Yes, optional	Yes, optional	Yes, optional	Yes, optional
<b>ACLs</b>	Yes, optional	Yes, optional	Yes, optional	Yes, optional
<b>Client checks</b>	Yes, optional	Yes, optional	Yes, optional	Yes, optional
<b>Network access webtop</b>	Yes (or full)	No	No	No
<b>Portal access webtop</b>	No	Yes (or full)	No	No
<b>Full webtop</b>	Yes	Yes	Yes	No

**Table 1.1** Configuration elements for Access Policy Manager access types

Figure 1.1 shows the configuration flow for the four types of access on Access Policy Manager.



**Figure 1.1** Configuration objects in Access Policy Manager

A client system can only connect using one of these configuration types at a time. However, you can configure multiple access types, and Access Policy Manager can dynamically determine the access type to provide during the access policy process, after the session starts.

The following sections describe each access type and scenario.

## Working with network access

**Network access** provides a full encrypted VPN tunnel from the client system to back end servers. Network access virtually puts the client machine inside the company network, so that clients perform operations exactly as if they sat within the corporate LAN. The administrator can configure access control lists that restrict access over the tunnel. Network access can provide

connections that are always available to supported clients. Typically, you use full network access as the deployment method for client computers that are from well-known or trusted sources, such as company-provided laptops.

For more information, see the ***BIG-IP® Access Policy Manager® Network Access Guide***.

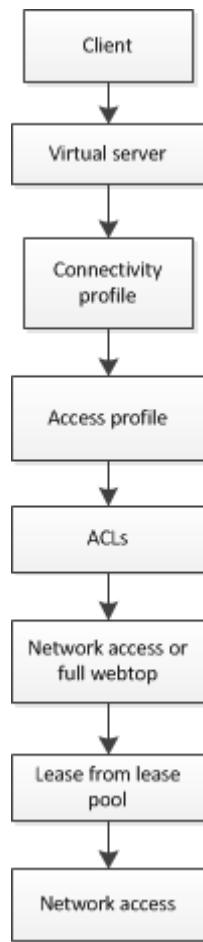
## Understanding a basic network access scenario

This basic network access configuration assigns a webtop and a connection to network access clients, and uses access control lists (ACLs) to control the resources and protocols a user can work with. This network access connection specifies no authentication.

In this access scenario, you define these objects:

- a connectivity profile
- a network access or full webtop
- a lease pool
- a network access resource
- one or more ACLs
- an access profile and an access policy that assigns the network access resource, network access or full webtop, and the ACLs
- a virtual server that specifies particular network access settings, including the connectivity profile and access profile

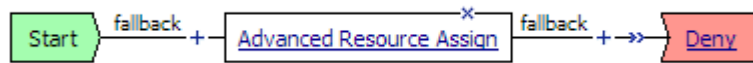
The objects that define this simple network access scenario are related as shown in Figure 1.2, on page 1-8.



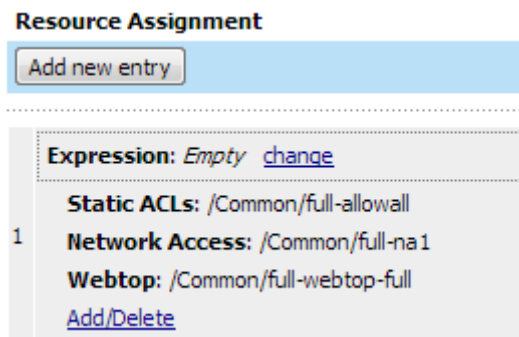
**Figure 1.2** Basic network access configuration object flow

The access policy for this scenario is very simple, and contains only one item: an advanced resource assign action that assigns the network access resource, the network access or full webtop, and any ACLs. The access policy is shown in Figure 1.3. An example resource assign action for this policy is shown in Figure 1.4.





**Figure 1.3** Basic network access configuration access policy



**Figure 1.4** Resource assign action configured for network access, an ACL and a full webtop

## Working with portal access

**Portal access** connections configure a remote access connection to one or more internal web applications. With this access type, users can access internal web applications through a single external virtual server. The portal access resource provides secure interaction with proprietary and standard web applications, using link rewriting technology. Typically, you use portal access on less trusted devices, or when full network access is not supported on a particular type of device. Use this if you need to provide secure extranet access to internal web-based applications without creating a full VPN connection.

For more information, see the *BIG-IP® Access Policy Manager® Portal Access Guide*.

## Understanding a basic portal access scenario

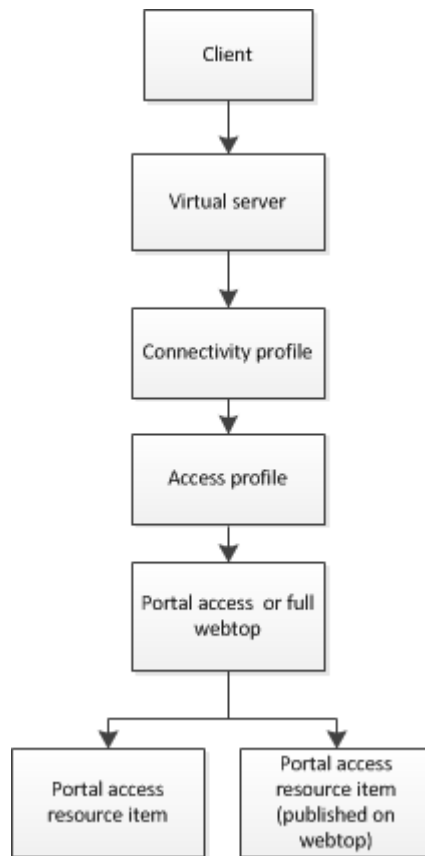
This basic web applications configuration assigns a webtop and portal access resource for use by a remote access user. This portal access configuration specifies no authentication.

In this access scenario, you define these objects:

- a portal access or full webtop
- a portal access resource and resource items
- a connectivity profile

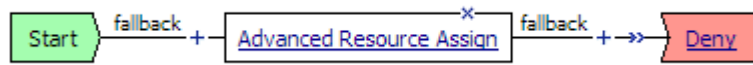
- an access profile and an access policy that assigns the portal access resource and the portal access or full webtop
- a virtual server that specifies particular portal access settings, including the rewrite profile, the connectivity profile, and the access profile

The objects that define this simple web applications scenario are related as shown in Figure 1.5.



**Figure 1.5** Basic web applications configuration object flow

The access policy for this scenario is very simple, and contains only one item: an advanced resource assign action that assigns the portal access resource, and ACL, and a full webtop. This access policy, as it appears in the visual policy editor, is shown in Figure 1.6. An example resource assign action for this policy is shown in Figure 1.7.



**Figure 1.6** Basic portal access configuration access policy

Properties **Branch Rules**

Name:

---

**Resource Assignment**

---

Expression: *Empty* [change](#)

1 Static ACLs: /Common/full-allowall

Portal Access: /Common/common-webtop-owa

Webtop: /Common/full-webtop-full

[Add/Delete](#)

**Figure 1.7** Resource assign action configured for portal access, an ACL and a full webtop

## Working with application access

**Application access** connections provide secure, application-level tunnel or remote desktop connections from the client to the network.

Additionally, optimization is available for app tunnels. Typically, you use application tunnels or remote desktop connections for users who require optimized access to applications or remote desktops.

For more information, see the **BIG-IP® Access Policy Manager® Application Access Guide**.

## Understanding a basic application access scenario

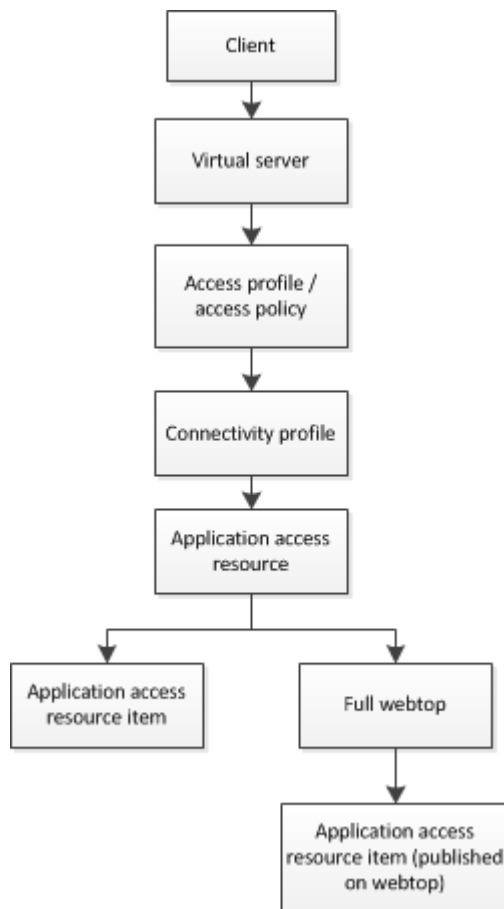
This basic application access configuration assigns an app tunnel resource for use by a remote access user. This application access configuration specifies no authentication.

In this access scenario, you define these objects:

- a full webtop
- an app tunnel resource and resource items
- a connectivity profile

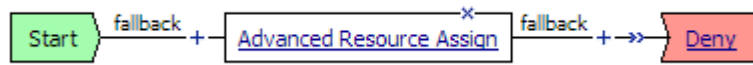
- an access profile and an access policy that assigns the app tunnel resource and the full webtop
- a virtual server that specifies particular app tunnel settings, including the connectivity profile and the access profile

The objects that define this simple application tunnel scenario are related as shown in Figure 1.8.



**Figure 1.8** Basic application access object flow

The access policy for this scenario is very simple, and contains only one item: an advanced resource assign action that assigns the application access resource, an ACL, and a full webtop. This access policy, as it appears in the visual policy editor, is shown in Figure 1.9. An example resource assign action for this policy is shown in Figure 1.10.



**Figure 1.9** Basic application access configuration access policy

The screenshot shows the configuration interface for the 'Advanced Resource Assign' action. At the top, there are tabs for 'Properties\*' and 'Branch Rules'. Below the tabs, the 'Name' field is set to 'Advanced Resource Assign'. Under the 'Resource Assignment' section, there is a button labeled 'Add new entry'. Below this, a list of resource assignments is shown, each with a label and a path:

- Static ACLs:** /Common/full-allowall
- App Tunnel:** /Common/at-http
- Remote Desktop:** /Common/rd-msrdp-crete
- Webtop:** /Common/full-webtop-full

At the bottom of the list is a link labeled 'Add/Delete'.

**Figure 1.10** Resource assign action configured for application access, an ACL and a full webtop

## Working with web access management

**Web access management** provides client-side security, authentication services, and access control to Local Traffic Manager virtual servers that load balance web applications. Typically, you use web access management to secure access to applications from a client system that is within a corporate environment.

For more information, see Chapter 2, .

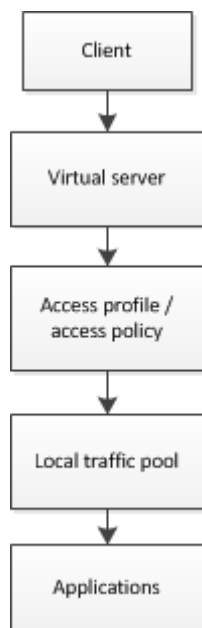
## Understanding a basic web access management scenario

This basic web access management configuration provides access control to a local traffic virtual server, and specifies client-specific ACLs. This access policy specifies no authentication.

In this access scenario, you define these objects:

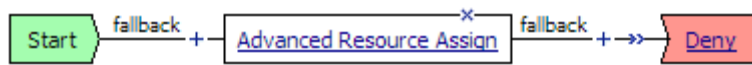
- a Local Traffic Manager virtual server with a configured pool
- an access profile and an access policy. The access profile is then selected in the Local Traffic Manager virtual server

The objects that define this simple web access management scenario are related as shown in Figure 1.11.



**Figure 1.11** Basic web access management object flow

The access policy for this scenario contains a start point, an ACL assign action, and an allow ending. You assign one or more ACLs to the access policy with the ACL assign action, and by doing so you control access to the local traffic management virtual server. For a web access management connection, no network access, portal access, app tunnel, or remote desktop resource is assigned, and no webtop is assigned. This access policy appears in the visual policy editor as shown in Figure 1.12. An example resource assign action for this policy, with only an ACL assigned, is shown in Figure 1.13.



*Figure 1.12 Basic web access management policy with ACLs*

The screenshot shows the configuration interface for an ACL Assign action. It has two tabs: 'Properties' and 'Branch Rules'. The 'Name' field is set to 'ACL Assign'. Below this, the 'ACL Assignment' section is visible, showing 'Static ACLs (1)' with a link to 'Add/Delete'. A single ACL is listed: '/Common/full-allowall'.

*Figure 1.13 ACL assign action for web access management, configured for an ACL only*

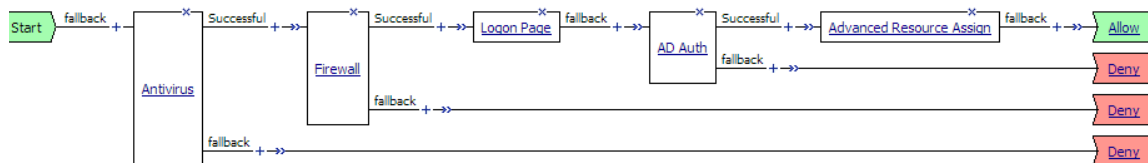
## Using access profiles and policies

Access policies are configured visually in the visual policy editor. In the visual policy editor, all access policies start with a start point, and every access policy has at least one rule branch. All access policies have one or more endings. A successful ending is an allow ending, and an unsuccessful ending is a deny ending. Between the start and the end point are access policy items, which define the behavior of the access policy. The access policy is similar to a flow chart, where you read flow of a user policy from left to right.

The simplest successful web access management policy has a start point, one or more ACLs, and an allow ending. This scenario, described in the section *Understanding a basic web access management scenario*, on page 1-14, provides access control features for a local traffic virtual server.

The simplest access policy includes a start point and an allow ending, and includes a resource assign action that assigns a connection resource and a webtop. When a user connects with this access policy, the user is assigned a connection and a webtop by the resource assign action. The user then goes to an allow policy ending, and the remote connection type is assigned to the user. Two such scenarios are described in the previous sections, *Understanding a basic network access scenario*, on page 1-7, and *Understanding a basic portal access scenario*, on page 1-9.

However, you typically check for client integrity, and require authentication to access resources, so a more typical access policy is shown in Figure 1.14. This access policy contains one or more client-side checks, such as antivirus, firewall, or operating system checks, a logon page and authentication action, and an advanced resource assignment action, followed by at least one allow ending, and deny endings for non-successful rule branches. The resource assignment action is used to assign connection resources, a webtop, webtop links, and any ACLs that apply to the connection. For a web access management connection, you can assign ACLs with the full resource assignment action or with the ACL assign action, but you do not assign a webtop, portal access, network access, or application access resources.



**Figure 1.14** A typical access policy in the visual policy editor

The basic access policy in Figure 1.14 includes actions that have successful and fallback rule branches (**Antivirus**, **Firewall**, **Active Directory authentication**), and actions that have single rule branches (**Logon Page** and **Advanced Resource Assign**).



You select an access profile in a virtual server definition, and the access policy associated with that access profile starts when a client connects to the virtual server. Access Policy Manager creates a blank access policy for every access profile. You can configure the access policy to dynamically assign objects to the user when the session starts, to determine the resources a user connects to, and to perform authentication and check client integrity. You can add logic and functionality to the access policy using configurable access policy items, and configure branches that change the flow of the policy. You can specify one or more connection resources and a webtop for the user as well.

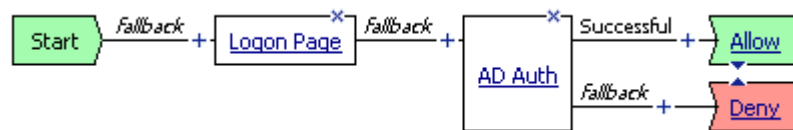
For more information on access policy structure and configuration, see Chapter 4, *Understanding Access Policies*, and Chapter 5, *Creating Access Profiles and Access Policies*.

## Using authentication in access policies

You can add authentication to an access policy using AAA servers (Authentication, Authorization, and Accounting) or client certificates.

Typically, you add two access policy items to add server authentication: a logon page action, and a AAA server action. Add the logon page action before the AAA server action. The logon page action presents a user with a logon page with customizable fields and text. The user enters credentials (for example, a logon name and password), and these credentials are then passed to the AAA server selected in the AAA server action. If a user is successfully authenticated, that user continues on the successful branch. A user who is not successfully authenticated continues on the fallback branch.

Figure 1.15 shows an access policy for web access management that includes authentication. This access policy includes only two items: a logon page action, and an Active Directory authentication action. This policy requires a user to authenticate successfully to Active Directory to connect to a local traffic virtual server, which is load-balancing applications.



**Figure 1.15** Simple access policy for web access management

## Assigning authentication in an access policy

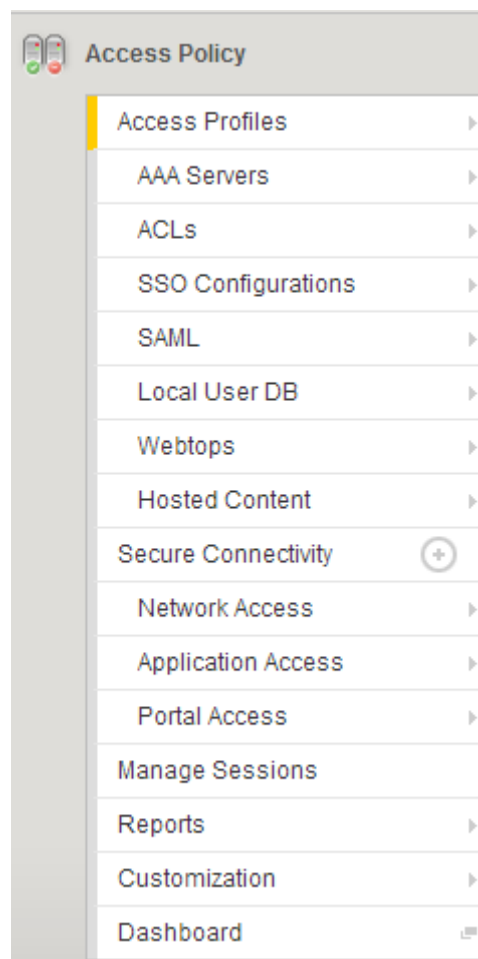
You can add authentication to any access policy or any branch in an access policy. You can even add multiple authentication types, so, for example, a user who fails Active Directory authentication might then attempt RADIUS authentication. You can configure multiple types of authentication, for

example, requiring users to authenticate with a certificate and with a AAA server. For more information on authentication methods and scenarios, see ***BIG-IP® Access Policy Manager® Authentication Configuration Guide***, and Chapter 9, *Using Certificate Authentication in APM*.

## Using the Configuration utility

The Configuration utility is the browser-based graphical user interface for the BIG-IP system. In the Configuration utility, the navigation pane main tab provides access to the access policy configuration objects, as well as the network, system, and local traffic configuration objects. The Help tab contains context-sensitive online help for each screen.

Figure 1.16 shows the Access Policy section of the navigation pane expanded.



**Figure 1.16** Access policy items in the Configuration utility navigation pane

## Overview of components of the Configuration utility

The Configuration utility contains the following components:

- ◆ **The identification and messages area**

The identification and messages area of the Configuration utility is the screen region that is above the navigation pane, the menu bar, and the body. In this area, you find the system identification, including the host name, and management IP address. This area is also where certain system messages display, for example Apply Access Policy, which appears when you need to activate an access policy.

- ◆ **The navigation pane**

The navigation pane, on the left side of the screen, contains the Main tab, the Help tab, and, the About tab. The Main tab provides links to the major configuration objects. The Help tab provides context-sensitive help for each screen in the Configuration utility. The About tab provides a quick way to view commonly used configuration objects.

- ◆ **The menu bar**

The menu bar, which is below the identification and messages area, and above the body, provides links to the additional configuration objects within each major object.

- ◆ **The body**

The body is the screen area where the configuration settings display.

## Getting started with BIG-IP Access Policy Manager

The Access Policy Manager® is a multi-featured appliance whose interface allows configuration from any location. To initially set up the secure access connections for users, you can follow different choices in your approach. We recommend setting up a basic working policy, using the Access Policy Manager connection wizards. To set up connections with the wizards, review the section *Using Access Policy Manager configuration wizards*, on page 1-21. You can follow the guidelines in *Following the recommended configuration path* section to set up Access Policy Manager, or you can elect to travel your own path, choosing from the options described in *Possible configuration scenarios*, on page 1-26.

### Using Access Policy Manager configuration wizards

With the Access Policy Manager® wizards, you can quickly configure any of the three access types with a simple working configuration. After you configure a connection with the wizard, you can go back and edit the configuration to further customize the access policy.

To access Access Policy Manager wizards, in the navigation pane, expand **Templates and Wizards**, and click **Device Wizards**. The Device Wizards screen opens.

The following wizards are available.

- **Network Access Setup Wizard for Remote Access** - Configures a working VPN connection. Typically, this allows users outside your network to connect to specified networks, and use their applications and network sites as if they are physically on the network.
- **Portal Access Setup Wizard** - Configures access to specific web applications for remote users. Typically, this allows users outside the network to connect to specified web applications, such as Microsoft Outlook Web Access or SharePoint, without allowing full access to the entire network.
- **Web Application Access Management for Local Traffic Virtual Servers** - Configures access to a local traffic virtual server managing web applications (web access management). Typically, this allows you to control access to the applications managed by the local traffic virtual server, using the features provided in the access policy. As an example, you can configure AAA server authentication, endpoint security, and other system checks before you allow access to the local traffic virtual server. You can configure this access type for an existing local traffic virtual server, or you can configure the virtual server with the wizard.

---

◆ **Note**

*The system includes online help for every screen in the wizard. To view the online help, click the Help tab in the navigation pane.*

## Using the network access wizard

Follow the steps and instructions in the wizard to configure and deploy a working network access connection. Note the following configuration items.

- The **Policy Name** specifies the name of the access policy to be created, and is used as the naming prefix for other objects configured with the access policy. Later, when you look for items created with the wizard, they are named with this prefix. For example, if you specify the prefix **mytest**, the access policy name is **mytest\_ap**, and the virtual server is named **mytest\_vs**. This name must be unique, and not already in use on the system.
- When you select the client side check option **Enable Antivirus Check in Access Policy**, the wizard adds a basic antivirus client-side check to the access policy. You can later refine this client-side check to verify a particular antivirus product, check the date of the virus database, and more. You can also add other client-side checks to the access policy. For more information, see Chapter 7, *Configuring Endpoint Security (Client-Side)*.
- You can configure authentication with the wizard, or select **No Authentication** to create an access policy without authentication. After you select an authentication type, you can view online help for the authentication configuration options by clicking the Help tab in the navigation pane.
- Lease pools are a configuration requirement for network access connections. Each connection is assigned an IP address from the lease pool. You must configure a lease pool with as many IP addresses as connected users you expect to host.
- Client settings can be configured for the connection with the wizard. We strongly recommend you read the **BIG-IP® Access Policy Manager® Network Access Guide**, and use the online help, if you plan to use settings other than the default values.
- DNS hosts for network access are required for your users to have functioning name resolution and Windows networking on your internal network. Specify a primary name server at a minimum. If you are using Microsoft networking features on your network, specify a primary WINS server.
- Specify a host name for the virtual server. In most cases, you do not specify a network when creating this virtual server. Allow the redirect server to be created; this eliminates the simple connection issue that users encounter when they do not type **https** before the virtual server host name.
- When you review the configuration, you can use the **Previous** and **Next** buttons to go back and edit the configuration before you click **Finish**. After you click **Finish**, the system creates and applies network access objects. You can still edit any item associated with the access profile from the Access Profile page (**Access Policy > Access Profiles > name of access profile**). You can edit the virtual server on the Virtual Server page (**Local Traffic > Virtual Servers > name of virtual server**).

## Using the portal access wizard

Follow the steps and instructions in the wizard to configure and deploy a working web applications access policy. Note the following configuration items.

- The **Policy Name** specifies the name of the access policy to be created, and is used as the naming prefix for other objects configured with the access policy. Later, when you look for items created with the wizard, they are named with this prefix. For example, if you specify the prefix **mytest**, the access policy name is **mytest\_ap**, and the virtual server is named **mytest\_vs**. This name must be unique, and not already in use on the system.
- When you select the client side check option **Enable Antivirus Check in Access Policy**, the wizard adds a basic antivirus client-side check to the access policy. You can later refine this client-side check to verify a particular antivirus product, check the date of the virus database, and more. You can also add other client-side checks to the access policy. For more information, see Chapter 7, *Configuring Endpoint Security (Client-Side)*.
- You can configure authentication with the wizard, or select **No Authentication** to create an access policy without authentication. After you select an authentication type, you can view online help for the authentication configuration options by clicking the Help tab in the navigation pane.
- Select the application from the list. You can select:
  - Domino Web Access (DWA)
  - Outlook Web Access (OWA) 2003, 2007 or 2010
  - Custom
- Select **Configure SSO** to set up Single Sign-On for web applications, and specify the **SSO Method** by selecting from the list.
- Specify the internal portal access start URI. This specifies the URI of the first page that a user sees after passing the access policy. For example, **http://myintranet.siterequest.com** or **http://myintranet/owa**.
- Specify a virtual server IP address or host name. Allow the redirect server to be created; this eliminates the simple connection issue that users encounter when they do not type **https** before the virtual server host name.
- When you review the configuration, you can use the **Previous** and **Next** buttons to go back and edit the configuration before you click **Finish**. After you click **Finish**, the system creates and applies portal access objects. You can still edit any item associated with the access profile from the Access Profile page (**Access Policy > Access Profiles > name of access profile**). You can edit the virtual server on the Virtual Server page (**Local Traffic > Virtual Servers > name of virtual server**).

## Using the web application access management wizard

Follow the steps and instructions in the wizard to configure and deploy a working web access management access policy. Note the following configuration items.

- On the first screen of the wizard, you have the option to continue the wizard and either use an existing virtual server or create a new virtual server with basic settings. Alternatively, you can cancel the wizard and create a virtual server manually, then later restart the wizard and select that virtual server in the configuration.
- The **Policy Name** specifies the name of the access policy to be created, and is used as the naming prefix for other objects configured with the access policy. Later, when you look for items created with the wizard, they are named with this prefix. For example, if you specify the prefix **mytest**, the access policy name is **mytest\_ap**, and the virtual server is named **mytest\_vs**. This name must be unique, and not already in use on the system.
- When you select the client side check option **Enable Antivirus Check in Access Policy**, the wizard adds a basic antivirus client-side check to the access policy. You can later refine this client-side check to verify a particular antivirus product, check the date of the virus database, and more. You can also add other client-side checks to the access policy. For more information, see Chapter 7, *Configuring Endpoint Security (Client-Side)*.
- You can configure authentication with the wizard, or select **No Authentication** to create an access policy without authentication. After you select an authentication type, you can view online help for the authentication configuration options by clicking the Help tab in the navigation pane.
- If you are creating a virtual server in the wizard, specify a host name for the virtual server. In most cases, you do not specify a network when creating this virtual server. Allow the redirect server to be created; this eliminates the simple connection issue that users encounter when they do not type **https** before the virtual server host name.
- Specify a pool member IP address. This specifies the IP address for a new member of a default local traffic pool. When you create the virtual server, the wizard defines a new default pool with one member, defined by this IP address.
- When you review the configuration, you can use the **Previous** and **Next** buttons to go back and edit the configuration before you click **Finish**. After you click **Finish**, the system creates and applies virtual server objects. You can still edit any item associated with the access profile from the Access Profile page (**Access Policy > Access Profiles > name of access profile**). You can edit the virtual server on the Virtual Server page (**Local Traffic > Virtual Servers > name of virtual server**).



## Following the recommended configuration path

If you are new to the Access Policy Manager®, you can follow the path outlined in this section. This recommended path is designed to guide you through the most common operations, and includes references to other sections with related functionality.

- ◆ Determine client-system security requirements.  
For more information, see *Understanding endpoint security (client-side) checks*, on page 7-1.
- ◆ Identify the authentication mechanism.  
The Access Policy Manager supports external authentication. You can select from a number of authentication methods, depending on the security setup you employ. These include Active Directory, RADIUS, LDAP, OAM, RSA SecurID, OCSP, CRLDP, TACACS+ and certificate-based security.
  - If you are not sure which type of authentication you want, review the *BIG-IP® Access Policy Manager® Authentication Configuration Guide*.
  - If you already have an authentication mechanism in place and you want to use it for verifying user identity, you can read more in the *BIG-IP® Access Policy Manager® Authentication Configuration Guide*.
  - Configure network access resources with the applications and functionality you want to provide, or create app tunnels, remote desktops, and portal access resources for your users. For web access management applications, you do not create resources or webtops. For more information, you can review the configuration guides for network access, portal access, and application access, and Chapter 2, .
- ◆ Create ACLs for users.  
For more information, see Chapter 3, *Configuring Resources*.
- ◆ Create an access profile and access policy that you can associate with your virtual server, to give your clients secure access.  
For more information, see Chapter 5, *Creating Access Profiles and Access Policies*.
- ◆ Assign resources to users.  
For more information, see *Assigning resources*, on page 6-8.
- ◆ Test user connectivity.  
This is a good place to stop and test to make sure that users can connect to the Access Policy Manager. To do so, open a new browser window and log on using a logon account that you know exists.
- ◆ Create client SSL profiles for users.  
For more information, see *Configuring client SSL profiles*, on page 9-8.
- ◆ Define your virtual server. See Chapter 10, *Configuring Virtual Servers*.
- ◆ Create advanced access policies, for more complex secure access scenarios.  
For more information, you can review the content in Chapter 11, *Advanced Topics in Access Policies*, and in the **BIG-IP Module**

*Interoperability Implementations Guide.*

- ◆ Read sample how-to scenarios.  
For more information, see Appendix 15, *Access Policy Example*.

## Possible configuration scenarios

There are several ways you can begin the configuration process.

- ◆ **To authenticate users from an authentication server**  
If you have an authentication mechanism in place and you want to use it to verify user identity, you can read more in the *BIG-IP® Access Policy Manager® Authentication Configuration Guide*.
- ◆ **To gather information from client systems**  
If you want to specify requirements for client systems to determine authentication (whether to grant user access) and authorization (which resources to grant access to), you can read more in Chapter 7, *Configuring Endpoint Security (Client-Side)*.
- ◆ **To configure the resources, applications, and functionality you want to provide**  
If you prefer to start with the resources, applications, and functionality that you want to provide to your users, you can read more in Chapter 3, *Configuring Resources*, the *BIG-IP® Access Policy Manager® Network Access Configuration Guide*, and the *BIG-IP® Access Policy Manager® Portal Access Configuration Guide*.
- ◆ **To learn about logging with the Access Policy Manager**  
If you want to get a head start on understanding the ongoing operations and logging functionality provided with the Access Policy Manager, review content in Chapter 12, *Logging and Reporting*.
- ◆ **To set up certificates on the server**  
If you are ready to set up and install server certificates for the Access Policy Manager, read more in Chapter 9, *Using Certificate Authentication in APM*.
- ◆ **To see access policy examples**  
If you want exposure to sample policies with step-by-step examples, see Appendix 15, *Access Policy Example*, and Chapter 11, *Advanced Topics in Access Policies*.

## Finding help and technical support resources

You can find additional technical documentation about the Access Policy Manager® using the following resources:

- ◆ The ***BIG-IP® Systems: Getting Started Guide*** describes how to initially set up, configure, and license your BIG-IP system. Before you set up the Access Policy Manager for the first time, we recommend that you read this guide in its entirety to become familiar with the product features, and the procedures for provisioning and licensing features.
- ◆ **Release notes**  
Release notes containing the latest information for the current version of the Access Policy Manager are available on the F5 Networks Technical Support web site at <http://support.f5.com>. This site includes release notes for current and previous versions of the Access Policy Manager.
- ◆ **Online help for Access Policy Manager features**  
You can find help online for all screens on the Configuration utility. To open the context-sensitive help in the Configuration utility, click the Help tab in the left navigation pane.  
To get help on a screen in the visual policy editor, click the **Help** button.
- ◆ **F5 Networks Technical Support web site**  
The F5® Networks Technical Support web site at <http://support.f5.com>, provides the latest technical notes, answers to frequently asked questions, release notes and release note updates, and the AskF5™ Knowledge Base. You can also find all the guides in PDF format.

## Finding the Access Policy Manager software version number

When you work with F5 Networks Technical Support, you might need to have the version number of the Access Policy Manager® (APM®) software that is running on your platform. You can find the software version number in the Configuration utility. Expand **System** in the navigation bar, then click **Configuration**. The Device General properties screen presents the host name, software version number, and other information. This is an example of the Properties and Operations table.

Host Name	apm.siterequest.com
Chassis Serial Number	bip012345s
Version	BIG-IP 10.1.0 Build 1400.0 Final

**Table 1.2** *Properties and Operations table listing the version number*





# 2

---

## Configuring Web Access Management

---

- Introducing web access management
- Reviewing web access management options
- Configuring web access management



## Introducing web access management

The BIG-IP® Access Policy Manager® provides various methods to pass user traffic and control access to applications by creating traffic tunnels using network access or allowing access to specific web applications.

However, the flexibility of Access Policy Manager provides another method to perform access control to web applications configured as local traffic pool members. This method of access is referred to as ***web access management***.

When used with BIG-IP® Local Traffic Manager™, Access Policy Manager provides access policy features only.

For more information on BIG-IP® Local Traffic Manager features, refer to ***BIG-IP® Local Traffic Manager™: Concepts***.

## Understanding how web access management works

Web access management provides users the ability to access web applications, through a web browser, without the use of tunnels or specific resources. In this scenario the user is authenticated and checked by the access policy in Access Policy Manager, without defining a resource or webtop. For example, you can have a configuration with ACLs, security checks, and authentication.

Through this method of access control, the Access Policy Manager communicates with backend web servers, forwarding requests from the client to web servers within a local traffic pool.

In a typical web access management connection, access occurs through a rewriting engine that rewrites links and URLs to and from the client. Web access management eliminates the need for content rewriting, allowing access to the configured local traffic pool after the user passes through the access policy checks.

In cases where you want additional security to your web applications where the access occurs on your local environment, we highly recommended that you use Access Policy Manager with Local Traffic Manager to achieve this.

## Reviewing web access management options

There are some web access management configuration options you may want to consider before setting up this method for web access management.

- **Front-end SSL**

The decision to either use or not use SSL should be dictated by the level of security required. Applications that perform any form of authentication where passwords are transmitted openly, or where any information between the client and the virtual server must be secured, should use SSL. Additionally, where SSL is used by the backend web servers, it is best to configure SSL by the virtual server.

- **HTTP profile compression**

You can enable compression on the HTTP profile used by the virtual server. Use compression to provide a better end user experience, particularly where there is limited bandwidth or high latency between the virtual server and the client.

## Setting timeouts for web access management policy management

The web access management type does not have a logout mechanism, so you must configure a custom timeout option from the following choices. Web access management timeouts are set due to user inactivity.

The following timeout mechanisms are:

- **Cache and session control access policy item** - The cache and session control access policy item terminates a user session when it detects that the browser window is closed. You can also use the cache and session control action in an access policy, to provide inactivity timeouts to the user session. Use the **Terminate session on user inactivity** setting to configure the timeout for a web access management session. The cache and session control action is supported on Windows browsers only. For configuration information, see *Setting up Windows cache and session control*, on page 8-2.
- **Access Profile properties.** You can configure a timeout in the access profile.
  - The **Maximum Session Timeout** setting provides an absolute limit for the duration of the access policy connection, regardless of user activity. If you want to ensure that a user session is closed after a certain period of time, configure this setting. Note that this setting is configured in seconds.
  - The **Inactivity Timeout** setting terminates the session if there is no traffic flow in the specified amount of time. Note that this setting is configured in seconds. Depending on the application, you may not want to set the inactivity timeout to a very short duration, as many applications may cache user typing, and generate no traffic for an extended period. In this scenario, a session may time out when the application is still in use, but the content of the user input is not relayed back to the server.



For configuration information, see *Understanding access profile settings*, on page 5-1.

## Understanding other web access management considerations

You must consider the following configuration items when configuring web access management.

- **SSL matching**  
SSL should be used consistently on the virtual server, as it is used with the web server. In other words, if the web server uses SSL, the virtual server should use SSL.
- **Multi-host service**  
When you implement a service with multiple hosts, access through the virtual server for new requests causes the load balancing algorithm for the associated member pool to select a new server. This can cause problems if persistence to a particular host is required.

## Configuring web access management

Configuring for web access management requires that you configure both the BIG-IP® Local Traffic Manager and Access Policy Manager.

When you configure for this method of access, you create a virtual server that has one or more pool members and HTTP servers, and you attach an access policy to that virtual server. This access policy optionally provides endpoint security, authentication, and access control lists. Nodes and pools that represent the web applications associate with this virtual server.

### Important

---

*When you create an access policy, the policy cannot include a network access or portal access resource or webtop.*

Configuring for web access management requires these basic steps:

- Create an access profile
- Create nodes that represent the web servers
- Add nodes to the pool
- Create a virtual server

### To create an access profile

1. On the Main tab of the navigation pane, expand **Access Policy**, and click **Access Profiles**.  
The Access Profile screen opens.
2. Click the **Create** button.  
The New Access Profiles screen opens.
3. Specify the information for all the required parameters.
4. Add any checks and actions required to the access policy. You can assign an ACL with the resource assign action, but do not assign a webtop or a portal access or network access resource.

### To create nodes that represent web servers

1. On the Main tab of the navigation pane, expand **Local Traffic**, and click **Nodes**.
2. Click **Create**.
3. Enter an address for the node.
4. Repeat and create additional nodes for every web servers you want to represent.
5. Click **Finished**.

**To add nodes to a pool**

1. On the Main tab of the navigation pane, expand **Local Traffic**, and click **Pools**.
2. Click **Create**.
3. For each node created, add them to the pool as New Members.
4. Click **Finished**.

**To create a virtual server**

1. On the Main tab of the navigation pane, expand **Local Traffic**, and click **Virtual Servers**.
2. Click **Create**.
3. Type the name and address of the virtual server.
4. Select a service port
5. Select the **HTTP Profile** from the available options.  
The default profile, **http**, is usually sufficient, unless additional configuration options are needed.
6. Select the **SSL profile (Client)** setting.  
A client SSL profile is only required if you want to enable SSL from the client to the virtual server.
7. Select the **SSL profile (Server)** setting.  
A server SSL profile is only required if the pool members require SSL.
8. From the **Access Profile** list, select an access profile you created for web access management.
9. Click **Finished**.

**To select a pool**

1. On the Main tab of the navigation pane, expand **Local Traffic**, and click **Virtual Servers**.  
The Virtual Server List screen opens
2. Click the name of the virtual server.  
The Virtual Server Properties screen opens.
3. Click the Resources tab.
4. From the **Default Pool** list, select the local traffic pool.
5. Click **Update**.





# 3

---

## Configuring Resources

---

- Understanding resources
- Using access control lists
- Configuring dynamic ACLs
- Using webtops
- Using AD query with IPv6



## Understanding resources

With BIG-IP® Access Policy Manager®, you use resources to provide secure connection functionality to users. With Access Policy Manager, you configure a resource to allow access to a web application or a network access connection, or you configure an access control list to allow or deny access to clients with a network access, web applications, or web access management policies.

You use access control lists (ACLs), network access, portal access, app tunnels, and remote desktop resources along with webtops to provide functionality to clients. For a web access management policy, you assign ACLs, but you do not assign any other resources. You use ACLs to define allowed and disallowed networks, hosts, and protocols for users. With all resource-based policies, you can assign a full webtop to provide useful links to users who connect. You assign ACLs and webtops dynamically in an access policy, using one of several resource assign actions.

In this chapter you can learn how to use ACLs and webtops. To configure network access resources, see the ***BIG-IP® Access Policy Manager® Network Access Guide***. To configure portal access, see the ***BIG-IP® Access Policy Manager® Portal Access Guide***. To configure app tunnels and remote desktops, see the ***BIG-IP® Access Policy Manager® Application Access Guide***.

## Using access control lists

You use *access control lists*, or ACLs, to restrict user access to specified host and port combinations.

For an ACL to have an effect on traffic, at least one access control entry must be configured. In an access control entry, the only item that is required is the action. When you configure an ACL with an entry with only an action defined, that action becomes the default access control action for all traffic to which the ACL is applied.

ACL entries can work on OSI Layer 4, the protocol layer, OSI Layer 7, the application layer, or both. When you first create an access control entry, you can select whether the entry is for Layer 4, Layer 7, or for both.

You can use a Layer 4 or Layer 7 ACL with network access, web applications, or web access management connections, with the following configuration notes.

- With network access, you can use a Layer 7 ACL that is configured to provide access control for port **80** HTTP connections. However, if you want to provide access control for anything that is not on port **80**, you must create a second virtual server, configured with the IP address to which the ACL entry applies, and the default access profile, **access**.
- For HTTPS network access connections, you can use Layer 7 ACL entries only if the virtual server has the private key of the backend server.
- If you assign no ACLs to an access policy, the default behavior allows access. To restrict resources to only those you specify in an ACL, add an ACL entry configured to reject all connections at the end of the ACL entry list. The access policy will then reject any connection not matched by a previous entry.

The order you specify for ACLs and ACL entries determines their priority. Access Policy Manager tests ACLs and ACL entries in order, based on their priority in the respective list. Access Policy Manager test ACLs assigned only to the current session. You can reorder ACL entries and ACLs.

You assign ACLs dynamically in the access policy with the advanced resource assign action or with the ACL assign action, so ACLs apply only to clients that reach that action in the access policy. See *To assign an access control list with the advanced resource assign action*, on page 3-5, for more information.

---

◆ **Note**

*ACLs are not enforced on network traffic initiated from the server. Use SNAT automap or SNAT pool options in the network access configuration if you do not want servers to be able to initiate a connection to any client.*

---

## Creating static access control lists

You create a static access control list to provide or deny access to network resources.



### To create a static access control list

1. On the Main tab of the navigation pane, expand **Access Policy**, and click **ACLs**.  
The ACLs screen opens.
2. Click **Create**.  
The New ACL screen opens.
3. In the **Name** field, type a name for the access control list.
4. From the **Type** list, select **Static**.
5. In the **Description** field, you can add an optional description of the access control list.
6. From the **ACL Order** list, you can optionally determine in what order to add the new ACL.
  - Select **After** to add the ACL after a specific ACL, that you can then select.
  - Select **Specify** to type the specific number of the ACL in the list.
  - Select **Last** to add the ACL at the last position in the list.
7. Click the **Create** button.  
The ACL Properties screen opens.
8. In the Access Control Entries area, click **Add** to add an entry to the access control list.  
The **New Access Control Entry** screen appears.
9. From the **Type** list, select whether this is a Layer 4 (**L4**), Layer 7 (**L7**), or Layer 4 + Layer 7 (**L4+L7**) access control entry.
10. From the **Action** list, select the action for the access control entry. If you are creating a default access control list, complete this step, then skip to the last step in this procedure.  
Actions for the access control list entry are:
  - **Allow** - Permit the traffic.
  - **Continue** - Skip checking against the remaining ACL entries in this ACL, and continue evaluation at the next ACL.
  - **Discard** - Drop the packet silently.
  - **Reject** - Drop the packet and send a TCP RST message on TCP flows or proper ICMP messages on UDP flows. Silently drop the packet on other protocols.  
**Note:** If HTTP traffic matches a Layer 4 ACL, a TCP RST message is sent. The ACL Deny page is sent when traffic is matched and denied on a Layer 7 ACL.
11. In the **Source IP Address** field, type the source IP address.  
This specifies the IP address to which the access control list entry applies.

12. In the **Source Mask** field, type the network mask for the source IP address.  
This specifies the network mask for the source IP address to which the access control list entry applies.
13. For the **Source Port** setting, select **Port** or **Port Range**.  
This setting specifies whether the access control list entry applies to a single port or a range of ports.
14. In the **Port** field or the **Start Port** and **End Port** fields, specify the port or port ranges to which the access control list entry applies.  
To simplify this choice, you can select from the list of common applications, to the right of the **Port** field, to add the typical port or ports for that protocol.
15. In the **Destination IP Address** field, type the IP address to which the ACL controls access.
16. In the **Destination Mask** field, type the network mask for the destination IP address.
17. For the **Destination Ports** setting, select **Port** or **Port Range**.  
This setting specifies whether the access control list entry applies to a single port or a range of ports.
18. In the **Port** field or the **Start Port** and **End Port** fields, specify the port or port ranges to which the access control list entry applies.  
To simplify this choice, you can select from the list of common applications, to the right of the **Port** field, to add the typical port or ports for that protocol.
19. From the **Scheme** list, select the URI scheme for the ACL entry.  
You can select **http**, **https**, or **any**.  
**Any** matches either HTTP or HTTPS traffic.
20. In the **Host Name** field, type a host to which the ACL applies.  
  
The **Host Name** field supports shell glob matching. For example, you can use the asterisk wildcard (\*) to search for zero or more characters, and the question mark wildcard (?) to search for a single character. For example, the host entry **\*.siterequest.com** matches **siterequest.com** with any prefix. This entry matches **www.siterequest.com**, **mail.siterequest.com**, **finance.siterequest.com**, and any others with the same pattern.  
  
The ? matches only the single character represented by the question mark, so **n?t.siterequest.com** matches the hosts **net.siterequest.com** and **not.siterequest.com**, but not **neet.siterequest.com**, **nt.siterequest.com**, or **note.siterequest.com**.
21. In the **Paths** field, type the path or paths to which the ACL applies.  
You can separate multiple paths with spaces, for example, **/news /finance**. The **Paths** field supports shell glob matching. You can use the wildcard characters \* and question marks (?) to

represent single or multiple characters. You can also type a specific URI, for example, **/finance/content/earnings.asp**, or a specific extension, for example, **\*.jsp**.

22. From the **Protocol** list, select the protocol to which the ACL applies.
23. From the **Log** list, select the log level for this access control entry.  
When events of this type occur, the server records a log message.  
Options are:
  - **None** - log nothing.
  - **Packet** - log the matched packet.
24. Click **Finished**.

### **To assign an access control list with the advanced resource assign action**

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a rule branch of the access policy, click the plus sign ( **+>** ) to add an action.  
The Add Item popup screen opens.
4. On the Assignment tab, select **Advanced Resource Assign**, and click **Add Item**.  
The Advanced Resource Assign action popup screen opens.
5. Click **Add new entry**.  
A new resource assign entry appears in the popup screen.
6. To add one or more ACLs, click the **Add/Delete** link, select **Static ACLs** tab from the menu bar, then select the check fields for ACLs you want to assign, and clear the check fields for ACLs you do not want to assign.  
ACL assignment is optional.
7. Click **Update** to return to the Resource Assign popup screen.
8. Click **Save** to save the action.

### **To assign an access control list with the ACL assign action**

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.

2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a rule branch of the access policy, click the plus sign ( **+** ) to add an action.  
The Add Item popup screen opens.
4. On the Assignment tab, select **ACL Assign**, and click **Add Item**.  
The ACL Assign action popup screen opens.
5. To add one or more ACLs, click the **Add/Delete** link, then select the check boxes for ACLs you want to assign, and clear the check boxes for ACLs you do not want to assign.  
ACL assignment is optional.
6. Click **Save** to save the action.

## Access control list examples

The following examples show how to use ACLs to prevent access to servers, or to allow only certain types of traffic to access servers.

### Example: Reject all connections to a specific network

In this ACL example, all connections to a specific network at **192.168.112.0/24** are rejected.

#### To configure an ACL to reject all connections to a specific network

1. To create the access control list, follow the instructions at *To create a static access control list*, on page 3-3.
2. Configure the access control entries as follows.
  - **Source IP Address - 0.0.0.0** (note that when you leave an IP address entry blank, the result is the same as typing the address **0.0.0.0**).
  - **Source Mask - 0.0.0.0**
  - **Source Ports - All Ports**
  - **Destination IP address - 192.168.112.0**
  - **Destination Mask - 255.255.255.0**
  - **Destination Ports - All Ports**
  - **Protocol - All Protocols**
  - **Action - Reject**
3. Click **Finished**.

## Example: Allow SSH access to a specific host

In this ACL example, SSH connections are allowed to the internal host at **192.168.112.9**.

### To configure an ACL to allow SSH connections

1. To create the access control list, follow the instructions at *To create a static access control list*, on page 3-3.
2. Configure the access control entries as follows.
  - **Source IP Address** - 0.0.0.0
  - **Source Mask** - 0.0.0.0
  - **Source Ports** - All Ports
  - **Destination IP address** - 192.168.112.9
  - **Destination Mask** - 255.255.255.255
  - **Destination Ports** - Port 22 (or select SSH)
  - **Protocol** - TCP
  - **Action** - Allow
3. Click **Finished**.

## Example: Reject connections to specific file types

In this ACL example, all connections that attempt to open files with the extensions **DOC**, **EXE**, and **TXT** are rejected.

### To configure an ACL to reject connections to specific file types

1. To create the access control list, follow the instructions at *To create a static access control list*, on page 3-3. Create a Layer4 + Layer7 ACL.
2. Configure the access control entries as follows.
  - **Source IP Address** - 0.0.0.0
  - **Source Mask** - 0.0.0.0
  - **Source Ports** - All Ports
  - **Destination IP address** - 0.0.0.0
  - **Destination Mask** - 0.0.0.0
  - **Destination Ports** - All Ports
  - **Scheme** - http
  - **Paths** - \*.doc \*.exe \*.txt
  - **Protocol** - All Protocols
  - **Action** - Reject
3. Click **Finished**.

## Configuring dynamic ACLs

You can add a dynamic ACL anywhere in an access policy before the resources are assigned. To add a dynamic ACL, you must complete several steps first.

### Understanding dynamic ACLS

A dynamic ACL is an ACL that is stored in an LDAP, RADIUS, or Active Directory server. Because the dynamic ACL is associated with the user directory, ACLs can be assigned specifically per the user session.

The access policy extracts the dynamic ACL from a field on the AD, RADIUS, or LDAP server. When the extraction happens, the access policy Dynamic ACL action takes the variable in the specified format, and converts it to an ACL that is applied to the access policy branch.

### Understanding the F5 ACL format

Access Policy Manager® supports ACLs in an F5 ACL format, and in a subset of the Cisco ACL format. You specify the F5 ACL in an attribute field in an Active Directory, RADIUS, or LDAP server, and then specify that attribute in the Dynamic ACL action.

The F5 ACL format is specified with the following commands:

```
{ action [logging_options] context }
```

### Understanding F5 ACL actions

The dynamic ACL action specifies an action that the ACL takes on traffic that matches the ACL context. Available actions are:

- **allow** - allows the specified traffic
- **reject** - rejects the specified traffic and sends a TCP RST code to the initiator
- **discard** - silently drops the packets
- **continue** - skips checking against the remaining ACL entries in this ACL, and continues evaluation at the next ACL

### Understanding F5 ACL logging options

Logging options can optionally be specified after the action in the F5 ACL format:

- **log** - enables default logging for the ACL
- **log-packet** - writes packet-level logs to the packet filter log file
- **log-verbose** - writes verbose logs
- **log-summary** - writes summary logs

- **log-config** - writes configuration logs to the configuration log file

## Understanding F5 ACL context options

Context options specify protocols, addresses, networks, and ports for the ACL action.

## Understanding F5 ACL protocols

Specify the protocol that the ACL matches. Options are:

- **ip** - IP protocol traffic
- **http** - HTTP protocol traffic. Requires that you specify an HTTP or HTTPS URL in the ACL definition
- **udp** - UDP traffic only
- **tcp** - TCP traffic only

Use the examples included to specify addresses for each protocol.

## Understanding F5 ACL addresses

In the F5 ACL format, the addresses are the last item specified in the ACL definition. Addresses are specified in a pair separated by a space. The access policy attempts to match the first address in the pair against the host, and the second address in the pair against the destination. Addresses can be:

- **any[/mask][:port]** - matches any host or IP address, with an optional subnet mask or a port. (for example,  

```
{ allow tcp any 1.2.3.4 }
```

allows TCP traffic between any host and the destination IP address 1.2.3.4.  

```
{ allow tcp any/8 1.2.3.4 }
```

allows TCP traffic between any host within the subnet **255.0.0.0** and the destination IP address **1.2.3.4**.  

```
{ allow tcp any/8:8000 1.2.3.4 }
```

allows TCP traffic between any host within the subnet **255.0.0.0** on port **8000** and the destination IP address **1.2.3.4**.
- **IP address[/mask][:port]** - matches a specific IP address, with an optional subnet mask or a port. For example,  

```
{ allow 1.1.1.1 1.2.3.4 }
```

allows TCP traffic between the host IP address **1.1.1.1** and the destination IP address **1.2.3.4**.  

```
{ allow 1.1.1.0/16 1.2.3.4 }
```

allows TCP traffic between host IP addresses on the network **1.1.1.0** with the subnet mask **255.255.0.0** and the destination IP address **1.2.3.4**.  

```
{ allow 1.1.1.1:22 1.2.3.4 }
```

allows TCP traffic between the host IP address **1.1.1.1** on port 22 and the destination IP address **1.2.3.4**.

## Specifying an F5 ACL with the IP protocol

The following example shows how to specify an IP protocol address in F5 ACL format. The context word **ip** is followed with an address pair specification, optionally preceded by an IP protocol number.

```
{ allow ip 51 any 1.2.3.4 }
```

## Specifying an F5 ACL with the TCP or UDP protocol

The following examples show how to specify a TCP or UDP protocol address in F5 ACL format. The context word **tcp** or **udp** is followed with an address pair specification.

```
{ allow tcp any 1.2.3.4 }
```

```
{ allow udp any 1.2.3.4 }
```

## Specifying an F5 ACL with the HTTP protocol

The following examples show how to specify an HTTP protocol address in F5 ACL format. The context word **http** is followed with a host address, a destination address, and a URL. The URL specification supports wildcards with glob matching.

```
{ allow http any 1.2.3.4 https://www.siterequest.com }
```

```
{ allow http any 1.2.3.0/24 http://*.siterequest.com/* }
```

```
{ allow http any 1.2.3.0/24 http://*.siterequest.???/* }
```

## Understanding the Cisco ACL format

You can use the Cisco ACL format to specify dynamic ACLs. Cisco format attributes are stored in a RADIUS server in Cisco AV-Pairs. In the access policy, you specify the Cisco option in the Dynamic ACL action, and the attribute **session.radius.last.attr.vendor-specific.1.9.1** is configured automatically.

The ACL is specified at

```
http://www.cisco.com/en/US/products/sw/secursw/ps1018/products\_tech\_note09186a00800a5b9a.shtml.
```

You can also specify the prefix

```
ip:inacl#X=
```

where X is an integer number which is used as rule identifier.

The **log** and **log-input** keyword has been mapped with F5 **log-packet** format.

The following keywords are not currently supported: **tos**, **established**, **time-range**, **dynamic**, and **precedence**.



## Specifying Cisco IP ACLs

For IP protocol, the following specification is supported.

**{deny|permit} protocol source source-wildcard destination  
destination-wildcard [log|log-input]**

For example

```
ip:inacl#10=permit ip any any log
```

## Specifying Cisco TCP ACLs

For TCP protocol, the following specification is supported.

**{deny|permit} tcp source source-wildcard [operator [port]] destination  
destination-wildcard [operator [port]] [log|log-input]**

For example

```
ip:inacl#10=permit tcp any host 10.168.12.100 log
```

## Specifying Cisco UDP ACLs

For UDP protocol, the following specification is supported.

**{deny|permit} udp source source-wildcard [operator [port]]  
destination destination-wildcard [operator [port]] [log|log-input]**

For example

```
deny udp any any log
```

## Creating a dynamic ACL container

A dynamic ACL container provides an unconfigured ACL that you select in the Dynamic ACL action.

1. On the Main tab of the navigation pane, expand **Access Policy**, and click **ACLs**.  
The ACLs screen opens.
2. Click **Create**.  
The New ACL screen opens.
3. In the **Name** field, type a name for the access control list.
4. From the **Type** list, select **Dynamic**.
5. In the **Description** field, you can add an optional description of the access control list.
6. From the **ACL Order** list, you can optionally determine in what order to add the new ACL.
  - Select **After** to add the ACL after a specific ACL, that you can then select.
  - Select **Specify** to type the specific number of the ACL in the list.
  - Select **Last** to add the ACL at the last position in the list.
7. From the **Match Case for Paths** list, select **Yes** to match case for paths, and **No** to ignore path case.

8. Click the **Create** button.  
The ACL Properties screen opens.

You need not configure the dynamic ACL container. Later, you select the dynamic ACL container in the Dynamic ACL action.

## Adding a dynamic ACL to an access policy

You add a dynamic ACL to an access policy, then you specify either the Cisco-AV format or the F5 ACL format, the AD, RADIUS, or LDAP attribute, and the dynamic ACL container.

Note that you must add the Dynamic ACL action after an authentication or query action, to capture the authentication variables that contain the dynamic ACL specification.

### To assign a dynamic access control list with the Dynamic ACL action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a rule branch of the access policy, click the plus sign ( **+** ) to add an action.  
The Add Item popup screen opens.
4. On the Assignment tab, select **Dynamic ACL**, and click **Add Item**.  
The Dynamic ACL action popup screen opens.
5. To add one or more ACLs, click the **Add new entry** button.
6. To use an F5 ACL from an AD, RADIUS, or LDAP directory, select **Custom**. To use a Cisco AV-Pair ACL from a RADIUS directory, select **Cisco AV-Pair VSA**.
7. In the **Source** field, type the attribute from which the Dynamic ACL action extracts ACLs.  
  
If you are using Cisco AV-Pair VSA from a RADIUS server, the field is prepopulated with **session.radius.last.attr.vendor-specific.1.9.1**.
8. From the **ACL** list, select the dynamic ACL container.
9. From the **Format** list, select the format in which the ACL is specified.
10. To add another ACL entry, click the **Add new entry** button and repeat the procedure.

11. Click **Save** to save the action.

## Using webtops

When a user is allowed access by an access policy, that user is typically assigned a webtop. A **webtop** is the successful end point for an access policy branch. A full webtop also provides a customizable screen for the user that includes webtop links, and all resources assigned to the access policy branch, except the ACLs. You can also assign a portal access or network access webtop for those specific connection types.

You assign a webtop to the user session in a resource assign action in the access policy. Make sure that you assign the correct webtop type.

- You assign a network access webtop with a network access resources only.
- You assign a portal access webtop with portal access resources only.
- You assign a full webtop to include a network access resource, multiple portal access resources, multiple app tunnels, multiple remote desktop resources, and customizable webtop links.

Many settings for the webtop can be customized. To customize webtop settings, see the ***BIG-IP® Access Policy Manager® Customization Guide***.

### To create a webtop


1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Webtops**.  
The Webtop List screen opens.
2. Click **Create**.  
The New Webtop screen opens.
3. In the **Name** field, type the name for the webtop.
4. From the **Type** list, select whether the webtop is a network access portal access, or full webtop.
5. If you selected a network access or full webtop, select whether to automatically minimize the webtop to the system tray, by selecting or clearing the **Minimize To Tray** check box.

When you select this setting for a network access webtop, the webtop automatically minimizes to the tray. With a full webtop, the webtop minimizes to the system tray only after the network access connection is started.

6. If you selected a portal access webtop, in the **Portal Access start URI** field, type the URI for the web application.
7. Click **Finished** to complete the configuration.

### To assign a webtop

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.

2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a rule branch of the access policy, click the plus sign (  ) to add an action.  
The Add Item popup screen opens.
4. On the Assignment tab, select **Advanced Resource Assign**, and click **Add Item**.  
The Advanced Resource Assign action popup screen opens.
5. Click **Add new entry**.  
A new resource assign entry appears in the popup screen.
6. Click **Add/Delete**.  
The resource assign popup screen appears.
7. To specify a webtop for the connection, click the **Webtop** tab, and select a webtop to assign.
8. Click **Update** to return to the Advanced Resource Assign popup screen.
9. Click **Save** to save the action.

---

◆ **Note**

*You can also assign a webtop using the Webtop and Webtop Links Assign action. See Assigning resources, on page 6-8, for more information.*

## Using AD query with IPv6

When an AD server is configured with an IPv6 address in the Domain controller setting, AD query does not work. However, AD query with IPv6 address has been tested with the following layered virtual server approach.

1. In the AD server configuration, use the host name of the DC in the Domain Controller setting. Here is an example.

```
apm aaa active-directory /Common/AD-IPv6 {
  admin-encrypted-password ".(.5(1EhJfN\\<^FaLGC0Bt8CG0KMfR\\9;coEKdIm=5@32II"
  admin-name Administrator
  domain enterprise.lab.fp.mynet.com
  domain-controller win2008.enterprise.lab.fp.mynet.com
}
```

### ◆ Note

*In the previous example, the host name is win2008.enterprise.lab.fp.mynet.com.*

2. Update the system's global setting to include a remote host entry for the DC host name that was used in step 1 and map it to an IPv4 address as shown in this example.

```
sys global-settings {
  gui-setup disabled
  hostname bigip2mgmt.lab.fp.mynet.com
  mgmt-dhcp disabled
  remote-host {
    /Common/abc { addr 172.31.54.99
      hostname win2008.enterprise.lab.fp.mynet.com
    }
  }
}
```

3. Create a pool with the DC IPv6 address as a member as shown in this example.

```
ltm pool /Common/AD-IPv6-Pool {
  members {
    /Common/fd00:ffff:ffff:fff1:912e:cdfe:c884:2607.any {
      address fd00:ffff:ffff:fff1:912e:cdfe:c884:2607
    }
  }
}
```

4. Create a layered wildcard TCP virtual server as follows:
  - Destination IP: The IPv4 address that was used in step 2, that is, 172.31.54.99
  - Service Port: 0 (All ports)
  - SNAT Pool: Auto Map

- Default Pool (in Resources): Pool created in step 3, that is, /Common/AD-IPv6-Pool

See this example.

```
ltm virtual /Common/bigip2.lab.fp.mynet.com-tcp {  
  destination /Common/172.31.54.99:any  
  ip-protocol tcp  
  mask 255.255.255.255  
  pool /Common/AD-IPv6-Pool  
  profiles {  
    /Common/tcp { }  
  }  
  snat automap  
  translate-port disabled  
  vlans-disabled  
}
```

5. Create another layered virtual as in step 4, but for UDP traffic. (Set the protocol setting in the Virtual server configuration to UDP). See this example.

```
ltm virtual /Common/bigip2.lab.fp.mynet.com-udp {  
  destination /Common/172.31.54.99:any  
  ip-protocol udp  
  mask 255.255.255.255  
  pool /Common/AD-IPv6-Pool  
  profiles {  
    /Common/udp { }  
  }  
  snat automap  
  translate-port disabled  
  vlans-disabled  
}
```

With the above configuration setting, AD query should work with a IPv6 back end DC.







# 4

---

## Understanding Access Policies

---

- Introducing access policies
- Understanding access policy items
- Understanding access policy branch rules
- Understanding access policy branches
- Understanding access policy macros
- Introducing access policy endings
- Understanding session variables



## Introducing access policies

In an access policy, you define the criteria for granting access to various servers, applications, and other resources on your network.

Using an access policy, you can define a sequence of checks to enforce the required level of security on a user's system, before the user is granted access to servers, applications, and other resources on your network.

An access policy can also include authentication checks, to authenticate a user before the user is granted access to the network resources.

With an access policy you can perform four basic tasks:

- ◆ **Collect information about the client system**

You can use the access policy to collect and evaluate information about client computers. For example, you can check that the user is operating from a company-issued computer, what antivirus software is present on the machine, what operating system the computer is running, and other aspects of the client configuration. This is accomplished using both client-side checks and server-side checks in the access policy.

- ◆ **Use the authentication action to verify client security against external authentication servers**

The access policy allows you to check and evaluate authentication against an external authentication database or a certificate, to make sure the client system recognizes the user.

- ◆ **Retrieve user's rights and attributes**

You can use the access policy to retrieve extended information from authentication servers including LDAP or Microsoft Active Directory attributes, and use the information retrieved to assign different resources.

- ◆ **Grant access to resources**

With the access policy, you assign a network access resource after the client is authenticated.

## Understanding access policy items

An access policy is made up of five kinds of access policy items. These are:

- A start point
- One or more actions
- Branches
- Macros and macrocalls
- One or more endings

### Understanding the access policy start point

Every access policy begins at a start point. In the visual policy editor, this is a green rectangle with an angled right side, labeled **Start**, that has one fallback branch connected to it. You build the access policy starting on this fallback branch.

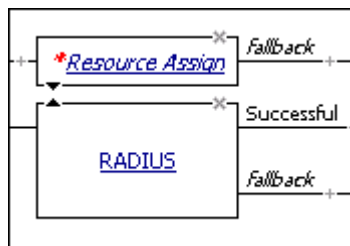


*Figure 4.1 An access policy Start point*

### Understanding access policy actions

An **action** performs a specific function in an access policy. These functions include client checks, authentication checks, and other access policy functions.

In the visual policy editor, the action appears as a rectangle surrounded by a single line in the access policy, with one branch entering it on the left, and one or more branches exiting on the right. If the action requires configuration, a red asterisk appears to the left of the action, and the name of the action appears in italics. In Figure 4.2, the RADIUS action is properly configured, and the resource assign action requires configuration.



*Figure 4.2 Two actions, one unconfigured, in the visual policy editor*

## Understanding available actions


The Access Policy Manager® (APM®) includes a number of pre-defined actions. You can see the available actions in the visual policy editor when you click the Add Item button , which is activated by positioning the cursor along the action's rule branch. The Add Item popup screen opens as a floating popup screen on top of the visual policy editor.

Table 4.1 lists all the actions available in Access Policy Manager, in the order in which they appear on each tab in the Add Item popup screen, and describes what they can do.

Category	Action	Description
Assignment	ACL Assign	Assigns static ACLs to the access policy branch.
	AD Group Resource Assign	Enables creation of a group name (named for an AD domain group) and resource assignment for the group.
	Advanced Resource Assign	Assigns ACLs, network access, portal access, app tunnels, and remote desktop resources. Also assigns a webtop and webtop links.
	BWC Policy	Adds a band-width control policy to a session.
	Citrix Smart Access	Enables Citrix Smart Access filters.
	Dynamic ACL	Assigns dynamic ACLs to the access policy branch.
	LDAP Group Resource Assign	Enables creation of a group name (named for an LDAP domain group) and resource assignment for the group.
	Pool Assign	Dynamically assigns a local traffic pool to a session. APM selects a pool in this priority order: <ul style="list-style-type: none"> <li>• a pool selected by an iRule that is defined for the virtual server takes precedence over any other</li> <li>• a static pool defined in the Pool Assign agent takes precedence over a static pool defined for the virtual server</li> </ul>
	Resource Assign	Assigns a network access resource, portal access, app tunnel, and remote desktop resources only.
	Route Domain and SNAT Selection	Dynamically selects a route domain and SNAT for policy-based routing.
	SSO Credential Mapping	Configures credential caching to use with single sign-on (SSO) for web applications.
	Variable Assign	Assigns one or more variables to the access policy.
	Webtop and Links Assign	Assigns a webtop and webtop links to the access policy branch.
Authentication	AD Auth	Adds Active Directory authentication to the access policy.

**Table 4.1** Available actions in Access Policy Manager

Category	Action	Description
Authentication	AD Query	Adds an Active Directory query to the access policy.
	Client Cert Inspection	If the Client SSL profile is configured to request the client certificate during the SSL handshake, checks the client certificate that was received during the SSL handshake.
	CRLDP Auth	Adds Certificate Revocation List Distribution Point (CRLDP) client certificate authentication to the access policy.
	HTTP Auth	Adds HTTP authentication to the access policy.
	Kerberos Auth	Adds Kerberos authentication, typically following an HTTP 401 Response action, to the access policy.
	LDAP Auth	Adds LDAP authentication to the access policy.
	LDAP Query	Adds an LDAP query to the access policy.
	LocalDB Auth	Adds authentication against a local user database instance to the access policy.
	NTLM Auth Result	Checks the result of NTLM authentication. (If NTLM authentication occurs, it happens before the access policy starts).
	OCSP Auth	Adds Online Certificate Status Protocol (OCSP) client certificate authentication to the access policy.
	On-Demand Cert Auth	If the Client SSL profile is configured to ignore the client certificate during the SSL handshake, reinitiates SSL handshake and prompts users for a client certificate.
	OTP Generate	Generates a one-time passcode.
	OTP Verify	Verifies a one-time passcode.
	RADIUS Acct	Adds RADIUS accounting to the access policy.
	RADIUS Auth	Adds RADIUS authentication to the access policy.
	RSA SecurID	Adds RSA SecurID two-factor authentication to the access policy.
	SAML Auth	Adds SAML authentication to the access policy.
	TACACS+ Acct	Adds sending accounting messages to a TACACS+ server when users log on and off.
	TACACS+ Auth	Adds TACACS+ Authentication of end user credentials to the access policy.

**Table 4.1** Available actions in Access Policy Manager

Category	Action	Description
Endpoint Security (Client-Side)	(Windows, Linux, Mac) File	Checks for a specific file on the client computer. File check is available as three different actions for Windows, Mac OS, and Linux computers.
	(Windows, Linux, Mac) Process	Checks for running processes on the client computer. Process check is available as three different actions for Windows, Mac OS, and Linux computers.
	Anti-spyware	Checks for anti-spyware software on the client computer. Can check for anti-spyware software on Windows and Mac systems.
	Antivirus	Checks for antivirus software on the client computer. Can check for antivirus software on Windows, Mac OS, and Linux clients.
	Firewall	Checks for firewall software on the client computer. Can check for firewall software on Windows, Mac OS, and Linux clients.
	Hard Disk Encryption	Checks for hard disk encryption software on the client compute (on Windows and Mac systems).
	Machine Cert Auth	Checks for the presence of a machine certificate.
	Patch Management	Checks for patch management software on the client computer (on Windows, Linux, and Mac systems).
	Peer-to-peer	Checks for peer-to-peer software on the client computer (on Windows, Linux, and Mac systems).
	Windows Cache and Session Control	Cleans and removes browser cache, and optionally cleans form entries, passwords, dial-up entries, and sets timeouts for the access policy.
	Windows Group Policy	Temporarily configures the Windows environment with a group policy. Windows Group Policy is an optional add-on that is enabled by FullArmor's GPAnywhere product.
	Windows Health Agent	Checks for Windows Health Agent software on the client computer on Windows.
	Windows Info	Checks for the version of Windows and for Windows updates on the client computer.
	Windows Machine Info	Collects Windows machine information from the client system, such as CPU, BIOS, network adapter, and hard disk details
	Windows Protected Workspace	Provides a secure computing environment with a temporary desktop and profile that is removed after logout. For use with public computers or in other situations where higher security is required.
	Windows Registry	Checks for specific values in the Windows registry.

**Table 4.1** Available actions in Access Policy Manager

Category	Action	Description
Endpoint Security (Server-Side)	Client for MS Exchange	Checks whether the client is a Microsoft Exchange client, such as Microsoft Outlook, and so on. This action requires an Exchange profile.
	Client OS	Detects the operating system of the remote client. Access Policy Manager detects this using information from the HTTP header.
	Client Type	Determines whether the user is connecting via a full or mobile browser, Edge Client®, Edge Portal®, Citrix Receiver, VMware View client, or Windows® Built-in Client. Note: Access Policy Manager can detect Windows Built-in Client only when the appropriate APM 11.4.x hotfix is installed. To determine hotfix requirements, refer to the BIG-IP APM Client Compatibility Matrix for APM 11.4.0 or APM 11.4.1 on the AskF5™ web site at <a href="http://support.f5.com">http://support.f5.com</a> .
	Client-Side Capability	Checks whether the client supports JavaScript and supports either ActiveX controls or Netscape plug-ins. If a client can support JavaScript and one of these control types, it can run client-side checks. See <i>Preparing for clients that cannot use client checks</i> , on page 8-1.
	Date Time	Branches based on date or time.
	IP Geolocation Match	Determines user's geographic location.
	IP Reputation	Checks reputation of client IP address.
	IP Subnet Match	Branches based on client's subnet.
	Jailbroken or Rooted Device Detection	Detects jailbroken or rooted mobile devices.
	Landing URI	Checks the landing URI that the client has used to start the current session.
General Purpose	License	Checks concurrent user license usage.
	Decision Box	Adds a decision box that provides two options for the access policy.
	Email	Configure Email messages for reporting.
	Empty	A blank action from which you can create your own action.
	iRule Event	Adds an iRule event to the access policy.
	Local Database	Allows read and write access to a local on-box user database.
	Logging	Adds a logging agent that logs the specified session variables to the system logs.

**Table 4.1** Available actions in Access Policy Manager



Category	Action	Description
Logon	Message Box	Adds a message box that can be used to post a message to the user.
	External Logon Page	Adds an external logon page to the access policy. Used with an external logon server like CSE's SECUREMATRIX.
Logon	HTTP 401 Response	Sends HTTP 401 Response for Basic or SPNEGO/Kerberos authentication.
	Logon Page	Adds a logon page to the access policy. You can customize the messages and link text on the logon page, and create custom messages for different languages.
	Virtual Keyboard	Displays a virtual keyboard on the logon screen when the user clicks in the Password box.
	VMware View Logon Page	Displays logon screen on VMware View clients. For more information, refer to <b>BIG-IP® Access Policy Manager® VMware Horizon View integration Implementations</b> .

**Table 4.1** Available actions in Access Policy Manager

## Understanding authentication actions

Authentication actions are used to add authentication with an authentication server or with a client certificate. Microsoft Active Directory and LDAP authentication actions can also be used to perform queries of the Active Directory or LDAP databases.

For more information on configuring authentication actions, see the *BIG-IP® Access Policy Manager® Authentication Configuration Guide*, and Chapter 9, *Using Certificate Authentication in APM*.

## Understanding access policy branch rules

A **branch rule** evaluates the result of an access policy action, findings about a client system, or other access policy item. The outcome of the evaluation of a branch rule grants or denies access, or continues on to the next action. The order of branch rules in an access policy determines the flow of action.

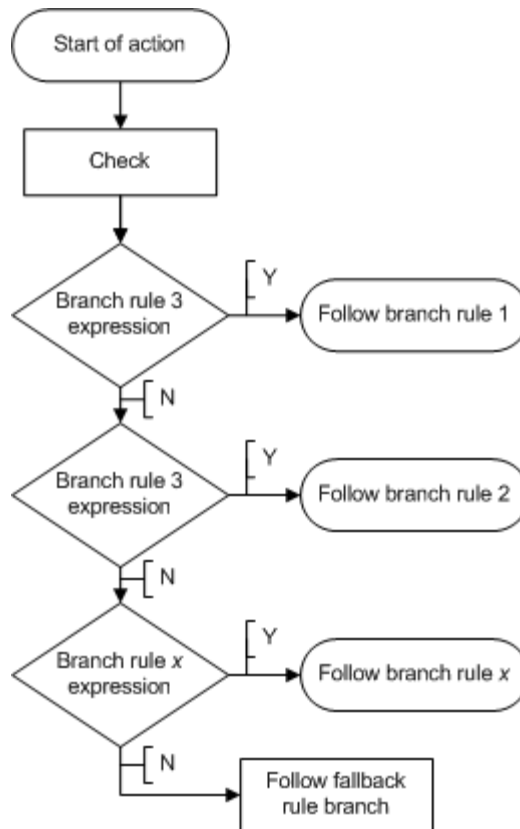
In an access policy, you use actions for which a set of branch rules are already defined. You can add branch rules to an action, or create new branch rules to test for a specific condition. You can use empty actions to create custom actions, and add your own branch rules to them. The ending is the last branch rule applied. Figure 4.3, on page 4-9, shows the flow of a branch rule-checking operation.

By default, if the user's system does not meet the access policy requirements, Access Policy Manager® denies the user access. You can change this outcome by changing the access policy ending, and by modifying branch rules to check for different criteria.

A branch rule uses data from variables returned by actions to determine user access criteria. For more information about session variables, see *Understanding session variables*, on page 4-19.

When you create a new action, the visual policy editor automatically creates a set of branch rules. The last rule in this set is the fallback branch rule. It cannot be moved. It governs all cases that do not satisfy a preceding branch rule.

Figure 4.3 shows the internal process of an action.



*Figure 4.3 Internal process of an action*

## Viewing rules


To view a predefined branch rule, you must first add an action to the access policy. The following example describes how to add a predefined action (client cert result) to an access policy, then how to view the underlying rule.

### ◆ Note

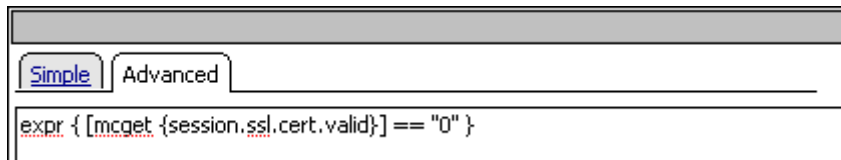
*You cannot view the predefined branch rules for every action.*

### To add a client cert inspection action and view the rule

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.

3. On a branch of the access policy, click the plus sign [  ] to add an action.  
The Add Item popup screen opens.
4. On the Authentication tab, select **Client Cert Inspection** and click **Add Item** to add the action to the access policy.  
The Client Cert Result action popup screen opens.
5. Click the **Branch Rules** tab.  
Under the Name **Successful**, you see the text  
**Expression: Client Certificate is valid**, and then a link to **change** the expression.
6. Click **change**.  
The Expression popup screen opens.
7. Click the **Advanced** tab.
8. The rule expression for the client cert result action is displayed, as in Figure 4.4:

**expr { [mcget {session.ssl.cert.valid}] == "0" }**



*Figure 4.4 A rule displayed in an access policy action*

## Predefined rules

When you configure an action, it creates a predefined rule. To further refine or customize a rule, you can use the expression builder to build a rule from a list of agents and conditions.

You can edit a rule on the **Branch Rules** tab by clicking **change**. You can edit rules in a rule builder on the Simple tab. You use this rule builder to choose from a simplified set of rules and automatically compile the Tcl syntax. You can also use the **Advanced** tab to edit the rule directly, using Tcl. Visual examples of the two editing methods are shown in Figure 4.5.

The image shows two screenshots of a rule editor interface. The top screenshot is in the 'Advanced' tab, showing a rule with the expression 'Client Certificate is valid'. Below this, there is an 'AND' operator and an 'Add Expression' button. The bottom screenshot is in the 'Simple' tab, showing a rule with the expression 'expr { [mcget {session.ssl.cert.valid}] == "0" }'.

Simple **Advanced**

Client Certificate is valid

AND

OR

**Simple** Advanced

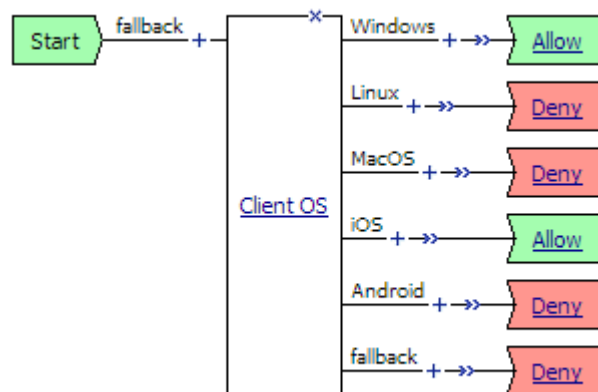
expr { [mcget {session.ssl.cert.valid}] == "0" }

**Figure 4.5** Simple (top) and Advanced (bottom) rule editing

## Understanding access policy branches

In the visual policy editor, you connect access policy items to other items with branches. A branch represents one of following three things:

- ◆ **The result of the evaluation of an access policy rule**  
Most actions have branches that represent the evaluation of rules. These branches might be called **Successful**, or they might have a more descriptive name. In many cases, a rule branch is a positive result to the evaluation of an action (for example, *Active Directory authentication has passed*). A rule branch can also be an informational response to the evaluation of an action (for example, *client operating system is Windows 7*).
- ◆ **An outgoing terminal from an access policy macro**  
When you configure an access policy macro, the rule branches inside the access policy macro have endings called terminals. These terminals do not function like access policy endings, but instead, become branches in the access policy to which the macrocall is added, which represent the outcomes of actions inside the macrocall.
- ◆ **A fallback rule**  
A fallback rule is typically a negative response, if the action has successful branches. Some fallback rules are the result of the action returning no match or a failure for the access policy check. Fallback rules are also the result of actions that have no positive or negative result. For example, the logon page action has no positive or negative result, because it sends only a logon page to the client, so the result branch of a logon page is always a fallback rule branch.

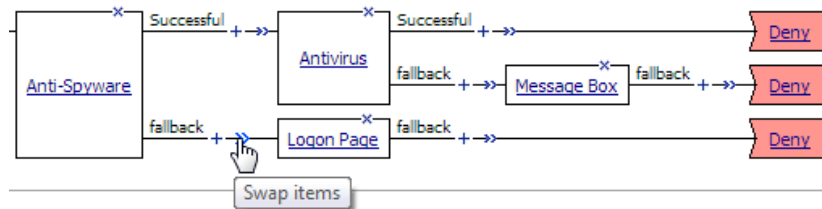


**Figure 4.6** An action with multiple branches

## About swapping access policy branches

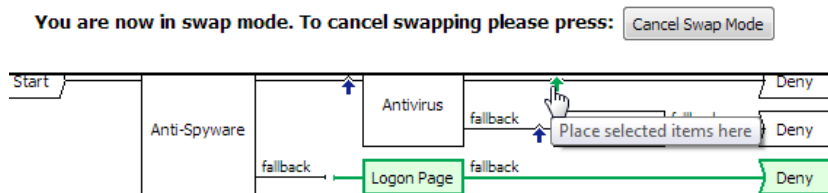
You can switch the contents of one access policy branch with another. Here is an example.

If you place your cursor over the swap [ →→ ] icon on an access policy branch, the **Swap items** tooltip displays.



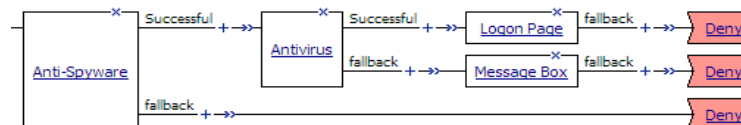
When you click the swap [ →→ ] icon on an access policy branch, the actions that are available to swap display in green and arrows display and point to possible branches with which to swap. When you place your cursor over an arrow, it too displays in green. Refer to Figure 4.7.

**Figure 4.7** Using swap mode in an access policy



After you click an arrow, the swap occurs. Figure 4.8 shows the result of the swap. The Logon Page and Deny ending (previously on the fallback branch after the Anti-Spyware action) switch places with the Deny ending on the Successful branch after the Antivirus action.

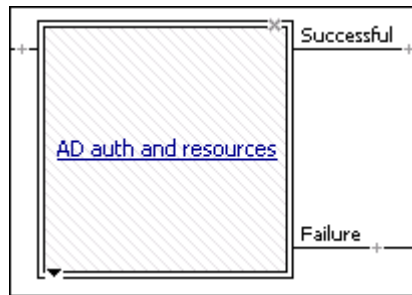
**Figure 4.8** Result of swapping branches



## Understanding access policy macros

A macro is a collection of actions that you can configure to provide common access policy functions. You can create a macro for any action or series of actions in an access policy. You can also create macros that contain macrocalls to other macros (nested macros).

After you create a macro, you place it in the access policy by adding an item called a macrocall to your policy. A **macrocall** is an action that performs the functions defined in a macro. In the visual policy editor, a macrocall appears in an access policy, or in a macro definition, as a single rectangular item, surrounded by a double line, with one or more outgoing macro terminal branches, called terminals, as shown in Figure 4.9

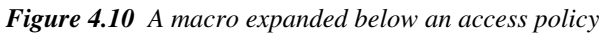


**Figure 4.9** A macrocall in an access policy

Macro definitions, macro terminals, and macrocalls are defined for each access policy. Macros you create in one policy do not appear, and cannot be used, in another access policy.

Unlike other access policy actions, when you click a macrocall in the access policy, the macro definition is displayed below the access policy in the macros section, and not in a popup screen, as shown in Figure 4.10.





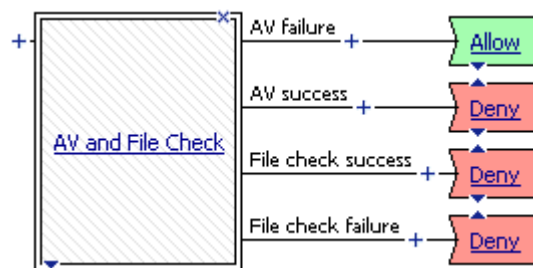
## Introducing macro terminals

Macro terminals are common shared endpoints for the access policy macro item. After you add a macro to the access policy using a macrocall, each macro terminal defined in the macro appears as a separate shared output. For example, if you configure four macro terminals, and use those terminals ten times in the macro definition, when you add the macrocall access policy item to the access policy, only four outputs appear from the access policy item.

For example, you can configure a macro with four terminals:

- AV success
- AV failure
- File check success
- File check failure

After you add the macrocall to your access policy, the macrocall appears as a single access policy item, with four terminals that appear as four branches, named for the terminals. See Figure 4.11.



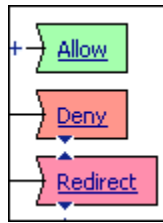
**Figure 4.11** A macrocall with four macro terminal branches in an access policy

◆ **Note**

*You can make changes to the actions in a macro after you have added the macrocall to an access policy. However, you cannot delete terminals after a macrocall has been added to an access policy or another macro. For this reason, we recommend that you configure macro terminals before you add a macrocall to the access policy.*

## Introducing access policy endings

Access policy endings indicate the final outcome of a branch of the access policy. The Access Policy Manager® provides the following endings: **Allowed**, **Deny**, and **Redirect**. In the visual policy editor, endings appear as a rectangle with a cut-out left edge.



*Figure 4.12 Access policy endings*

## Understanding the allow ending

In an access policy, the **allow ending** is a successful ending that allows the connection defined by the access policy branch. Configure your access policies so that only users who meet your security criteria reach an allow ending. The allow ending performs final validation of assigned resources, the webtop, and any resources added to the access policy branch, and allows the session to start.

### ◆ Note

*You must assign a valid network access or web application resource and a webtop for your users, unless you are using the access policy to control access to a local traffic virtual server, in a web access management scenario.*

## Understanding the deny ending

In an access policy, the **deny ending** denies the user access to the resource, and ends the user's session. After the user reaches a deny ending, all the session information collected during access policy operation is deleted from the client. You can use this ending at the ends of failed rule branches. When a user reaches a deny ending, the user sees an access denied error message web page.

## Understanding the redirect ending

In an access policy, the ***redirect ending*** sends the user to a URL that you specify. Use this ending when the result of a certain access policy outcome does not result in a webtop ending, but you want to send the user to another internal or external URL. For example, you might send a user to the web site for an antivirus vendor, if an antivirus action determines that the user's virus definitions are older than the access policy allows.

To close the Access Policy Manager session after the redirect, select the **Close session after redirect** check box.

---

◆ **Note**

*You must type the redirect URL with the leading **http://** or **https://**.*

## Understanding session variables

The rules in access policies use the values that the actions return in session variables. During access policy operation, the Access Policy Manager® collects various information about the system that is attempting access. This information is organized in a hierarchical arrangement and is stored as the user's session data.

**Session variables** are variables that allow the access policy to access user's session data. The name of a session variable consists of multiple hierarchical nodes separated by periods (.).

The Access Policy Manager names session variables in the following manner:

**session.ad.<username>.queryresult** = query result (0 = failed, 1=passed)

**session.ad.<username>.authresult** = authentication result (0 = failed, 1=passed)

**session.ad.<username>.attr.<attr\_name>** = the name of an attribute retrieved during the Active Directory query. Each retrieved attribute is converted to a separate session variable. Note that attributes assigned to a user on the AAA server are specific to that server, and not to Access Policy Manager.

Figure 4.13 shows how Access Policy Manager names session variables.



*Figure 4.13 Session variable naming scheme*

## Using session variables

You can use session variables to customize access rules or to define your own access policy rules. You can assign users specific resources based on session variables, using the resource assign action.

You can use session variables to configure rules in access policies. You can use the values of session variables to provide different outcomes for policies. For more information on how to use session variables, see *Assigning variables*, on page 6-10, and *Using advanced access policy rules*, on page 11-16. For a complete listing of available session variables, see Appendix 14, *Session Variables*. You can view all session variables for a session at **Reports > Current Sessions**. Click a session name to view the session variables for the session.



# 5

---

## Creating Access Profiles and Access Policies

---

- Creating an access profile
- Creating an access policy
- Configuring macros
- Exporting and importing access profiles





## Creating an access profile

In the BIG-IP® Access Policy Manager® an **access profile** is the profile that you select in a virtual server definition to establish a secured connection to a resource. You can also configure an access profile to provide access control and security features to a local traffic virtual server hosting web applications.

The access profile contains:

- Access policy timeout and concurrent user settings
- Accepted language and default language settings
- Single Sign-On information and domain cookie information for the session
- Exchange profile for Microsoft Exchange service settings  
An Exchange profile specifies SSO configurations for Exchange services.
- Customization settings for the access profile  
To customize these settings, see the *BIG-IP® Access Policy Manager® Customization Guide*.
- The access policy for the profile

## Understanding access profile settings

On the Access Profile Properties screen, you use the Settings section to configure timeout and session settings. You must select the **Custom** check box to configure settings for this section.

- **Inactivity Timeout** - Specifies the inactivity timeout for the connection, in minutes. If there is no activity between the client and server within the specified threshold time, the system closes the current session. By default, the threshold is **0**, which specifies that as long as a connection is established, the inactivity timeout is disabled. However, if an inactivity timeout value is set, when server traffic exceeds the specified threshold, the inactivity timeout is reset.  
In addition, for portal access, you can customize the timing for the warning message to appear for the user prior to session timeout by using the **Session Timeout Guard Time** setting in the webtop customization settings. The user can click a link inside the message window to reset inactivity timeout.
- **Access Policy Timeout** - This is designed to keep malicious users from creating a DOS attack on your Secure Access Manager. The timeout requires that a user, who has followed through on a redirect, must reach the webtop before the timeout expires. The default value is **300** seconds.
- **Maximum Session Timeout** - Specifies the maximum lifetime of one session, in minutes. The maximum lifetime is between the time a session is created, to when the session terminates. By default, it is set to **0**, which

means no limit. When you configure this setting, there is no way to extend the session lifetime, and the user must logout and then log back in to the server, when needed.

- **Max Concurrent Users** - Specifies the number of sessions per access profile. The default value is **0**, which represents unlimited sessions. Please note that this field is read-only for application editors. All other administrative roles can modify this field.
- **Max Sessions Per User** - Specifies the number of sessions per user. The default value is **0**, which represents unlimited sessions. Please note that this field is read-only for application editors. All other administrative roles can modify this field.

## Understanding configuration settings

On the Access Profile Properties screen, you use the Configurations section to configure a logout URI and timeout.

- **Logout URI Include** - Specifies a list of logoff URIs that the access profile searches for in order to terminate the APM session. This feature is used with http applications.
- **Logout URI Timeout** - Specifies the timeout used to delay logout for the customized logout URIs defined in the logout URI Include list.
- **Exchange** – Specifies an Exchange profile, which contains settings for Microsoft Exchange services.

## Understanding Single-Sign On settings

On the Access Profile Properties screen, you use the Single-Sign On settings to configure Single Sign-On and cookie behavior, with the following settings:

- **Domain Mode** - Two domain modes are available: Single Domain or Multiple Domains.
  - **Single Domain** - Select this if you want to apply your SSO configuration for only a single domain.
  - **Multiple Domains** - Select this if you want to apply your SSO configuration across multiple domains. This is useful in cases where you want to allow your users a single APM login session and apply it across multiple Local Traffic Manager or Access Policy Manager virtual servers front-ending different domains.
- **Primary Authentication URI** - Specifies the address of your primary authentication URI. This is a required field if you select to use SSO configuration across Multiple Domains. An example would be **https://logon.yourcompany.com**. This is where the user session is created. As long as you provide the URI, your user is able to access

multiple backend applications from multiple domains and hosts without requiring them to re-enter their credentials because the user session is stored on the primary domain.

- **Cookie Options** - This setting applies to Single Domain or the primary authentication domain. The following options are available:
  - **Secure** - Enable this setting to add the secure keyword to the session cookie. If you are configuring an application access control scenario where you are using an HTTPS virtual server to authenticate the user, and then sending the user to an existing HTTP virtual server to use applications, clear this check box.
  - **Persistent** - Enable this setting to set cookies if the session does not have a webtop. When the session is first established, session cookies are not marked as persistent, but when the first response is sent to the client after the access policy completes successfully, the cookies are marked persistent.  
Persistent cookies are updated for the expiration timeout every 60 seconds. The timeout is equal to session inactivity timeout. If the session inactivity timeout is overwritten in the access policy, the overwritten value will be used to set the persistent cookie expiration.
- **SSO Configuration** - To add an SSO configuration for Single Sign-On, select the configuration from the list.
- **Domain Cookie** - Applies to single domain mode only. Specifies a domain cookie to use with a web access management connection. If you specify a domain cookie, then the line **domain=specified\_domain** is added to the **MRHsession** cookie.  
By default, the **Secure Cookie** option is enabled. This adds the secure keyword to the session cookie. If you are configuring a web access management scenario with an HTTPS virtual server for authentication, and using an HTTP local traffic virtual server for applications, clear this check box.
- **Configure Authentication Domains** - The following options apply only if you select to use Multiple Domains.
  - **Domain/Host list** - Type in the domain or host that you want to apply the SSO configuration. Click Add to specify additional domains or hosts.
  - **Secure Cookie** - Enable this setting if you want your domain or host to add the secure keyword to the session cookie. If you are configuring an application access control scenario where you are using an HTTPS virtual server to authenticate the user, and then sending the user to an existing HTTP virtual server to use applications, clear this check box.
  - **Persistent Cookie** - Enable this setting if you want your domain or host to retain the cookie for the user session, even when the user session is terminated. Although this is an unsecure method, this setting is useful, and required, in cases where you have a third-party application, such as Microsoft SharePoint, and need to store the cookie in a local database so that any attempt to access backend server applications through Access Policy Manager succeeds.

- **SSO Config** - For each domain or host that you add, you can apply different SSO authentication methods. Select an existing SSO configuration from the list. Access Policy Manager supports different SSO mechanisms for different applications protected by a single access policy.

## Creating an access profile

### To create an access profile

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. Click **Create**.  
The New Access Profile screen opens.
3. In the **Name** box, type a name for the access profile.  
The Access Profile Properties screen appears.
4. To change settings for **Inactivity Timeout**, **Access Policy Timeout**, **Maximum Session Timeout**, and **Max Concurrent Users**, select the **Custom** check box, then type numbers for the settings you want to change.
5. To select a Single Sign On (SSO) configuration for the access policy, specify settings in the Single-Sign On section.
6. (Optional) In the **Domain Cookie** box, type the domain cookie.
7. Select the **Secure** check box in the Cookie Options section to add the secure keyword to the domain cookie.  
If the access policy is configured for an HTTP virtual server, clear this check box.
8. Configure the language settings for the access profile.  
See *Customizing access profile languages*, following, for more information.
9. Click **Finished** when the configuration is complete.

## Applying an access policy

After you create or change an access policy, the link **Apply Access Policy** appears in yellow at the top left of the BIG-IP® Configuration Utility screen. You must click this link to activate the access policy for use in your configuration.

### To apply access policies

1. Click the **Apply Access Policy** link.  
The Apply Access Policy screen appears, showing a list of access policies that have been changed.

2. Select the check boxes for one or more access policies to apply, and click the **Apply Access Policy** button.

By default, all access policies that are new or changed are selected.

After you apply the access policy, the Access Profiles list screen is displayed.

## Customizing access profile languages

Typically, the client's web browser has language preferences configured, which lists display languages in order of preference. Access Policy Manager® detects this order, compares it with the languages configured in the access profile, and presents customized pages and messages in the user-specified language, if that language exists in the access profile. If the user-specified language does not exist in the access profile, the user sees pages in the access profile default language.

In the access profile, you can configure the list of accepted languages in which the Access Policy Manager provides messages and customized elements. You can also select a default language for the access profile. The default language is used to provide messages and customized elements to users whose browsers are not identified with a language that is on the list of accepted languages.

Several languages have predefined messages in Access Policy Manager. Those languages appear in the **Factory Builtin Languages** list.

Languages that are available, but not yet customized for use with Access Policy Manager appear in the **Additional Languages** list.

There are several other places in Access Policy Manager where you can customize settings for different languages. To configure these language settings, see the following tasks and pages:

- *Customizing the Deny access policy ending*, on page 5-13
- *Customizing access profile languages*, on page 5-5

### ◆ Note

*If you customize messages, you must customize the same messages separately for each accepted language. Otherwise, default messages will appear for any accepted language for which you have not customized messages. It is recommended that if you customize messages for a specific accepted language, you remove all other languages from the accepted language list.*

### To customize access profile languages

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen appears.
2. In the Access Profiles List, click the name of the access profile you want to edit.

3. Configure the access profile language options as follows:
  - To add a language string from the list of installed languages, in the Language Settings area, in the **Factory Builtin Languages** box, select the language, and click the ( << ) button to move the language to the **Accepted Languages** list.
  - To add a language that is not yet installed and customized, from the **Additional Languages** list, select the language and click **Add**.
4. Click **Update** to update the language settings.

## Creating an access policy

In an access policy, you define the criteria for granting access to various servers, applications, and other resources on your network.

You create an access policy by creating an access profile, which automatically creates a blank access policy. Every access profile has an access policy associated with it. You configure that access policy through the access profile.

## Starting the visual policy editor

To view and edit the access policy associated with an access profile, you use the *visual policy editor*, a browser-based editor for access policies.

### To start the visual policy editor

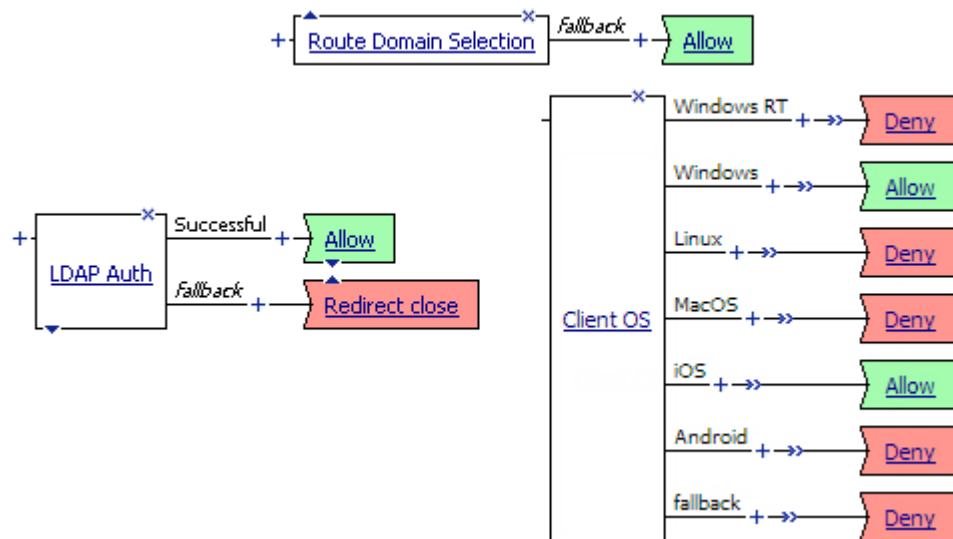
1. On the Main tab of the navigation pane, expand **Access Policy** and click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the Access Policy column click **Edit** for the access policy you want to edit.  
The visual policy editor opens in a new window or new tab, depending on your browser settings. You can right-click and select to open in a new tab or new window, if you want to choose the destination.  
If this is a new access policy, an unconfigured policy appears.

You can also open an access policy from the Access Profiles List screen by clicking the access profile name, then clicking the Access Policy tab, then clicking the **Edit** link.

## Using branch rules

In the visual policy editor, policy branch rules follow each policy action. Typically, an action is followed by both a successful branch rule and a fallback branch rule. Some actions, like the Logon action, are followed by only one branch rule. Some actions are followed by multiple branch rules. In actions where there is only one result branch rule, that result is labeled **Fallback**. In actions where there is a failed result and a successful result, the visual policy editor labels the successful branch rule **Successful** and the failed branch rule **Fallback**. Some actions have multiple result branch rules, and no successful branch.

For example, the Client OS action in Figure 5.1 has multiple branch rules, and each branch rule is named for the operating system to which the branch rule corresponds, with a fallback branch for any client operating system that does not match a specific branch rule. This allows you to assign actions to any branch rule, and separate endings to any branch rule.



**Figure 5.1** Policy actions with various result branch rules

#### ◆ Note

The Windows® RT branch shown in figure 5.1 is available only when you have the appropriate Access Policy Manager® 11.4.x hotfix installed. To determine hotfix requirements, refer to the BIG-IP® APM Client Compatibility Matrix for APM 11.4.0 or APM 11.4.1 on the AskF5™ web site at <http://support.f5.com>.

#### To add actions to a branch rule

Click the plus sign on the branch rule where you want to add the action. When you place your cursor over the plus sign, it turns blue and appears between parentheses [ (+) ] to indicate that you can click it.

## Configuring a basic access policy

To configure a basic access policy, you need to complete the following tasks.

- ◆ Create an access policy. For more information, see *Opening an access policy*.
- ◆ Add general purpose actions, client side checks, and server side checks, as needed.
- ◆ Add authentication. For more information, see *BIG-IP® Access Policy Manager®: Authentication Configuration Guide* on <http://support.f5.com>.



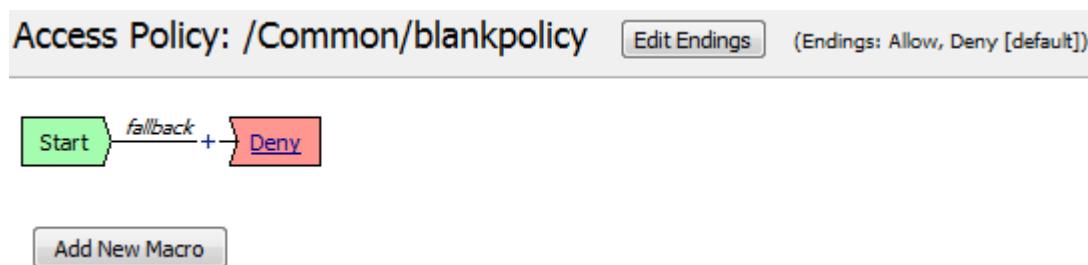
- ◆ Assign resources. For more information, see *Assigning resources*, on page 6-8.  
Note that you must assign a resource group that contains a network access resource, or the access policy will not function.
- ◆ Finish the access policy. For more information, see *Applying an access policy configuration*, on page 5-14.

## Opening an access policy

When you create an access profile, the system automatically creates an associated, blank access policy.

### To open an access policy

1. On the Main tab of the navigation pane, expand **Access Policy** and select **Access Profiles**.  
The Access Profiles List screen opens.
2. Click **Edit** in the Access Policy column of the access policy you want to edit.  
The visual policy editor opens, displaying the access policy.




*Figure 5.2 A new, unconfigured access policy*

## Adding actions to an access policy

When you first open a new access policy in the visual policy editor, the configuration includes only a start point, a fallback branch rule, and a default ending.

### To add an action to an access policy

1. On the Main tab of the navigation pane, expand **Access Policy** and click **Access Profiles**.  
The Access Profiles List screen opens.

2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a branch rule of the access policy, click the plus sign (  ) to add an action.  
The Add Item popup screen opens.
4. If the tab for the category of action that you want to add is not open, click another tab.
5. Select an action to add to the access policy.  
See the full list of action categories and actions at *Understanding available actions*, on page 4-3.
6. Click **Add Item** to add the action to the access policy.  
The action popup screen opens.  
To configure the action, see the action description at *Understanding available actions*, on page 4-3.

## Using policy endings

**Access policy endings** are the end result of a branch rule in an access policy. With access policy endings, you can give users access to the network access connection, deny access to users, or redirect users to another URL.

There are three types of endings:

- **Allow**  
Starts the SSL VPN session and loads the sources and webtop for the user.
- **Deny**  
Disallows the SSL VPN session and shows the user a Logon Denied web page.
- **Redirect**  
Transfers the user to the URL specified in the ending configuration.

## Configuring access policy endings

In the visual policy editor, you can create and delete access policy endings, change any ending in the access policy to another ending, customize endings, and set a default ending.


### To create an access policy ending

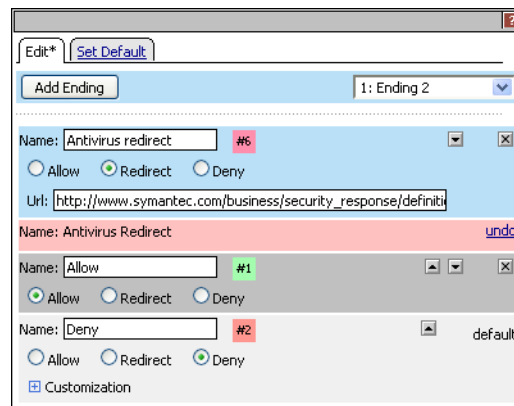
1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.

2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. Near the top of the visual policy editor, click the **Edit Endings** button.  
The Edit popup screen opens.
4. At the upper left, click the **Add Ending** button.  
The new ending appears, highlighted in blue. See Figure 5.4.
5. In the **Name** box, type a name for the new ending.
6. Select the type of ending (webtop, logon denied, or redirect).
  - **Allow**  
Specifies that the user has access to the VPN connection, as defined in the access profile and access policy.
  - **Redirect**  
Specifies a URL to which the access policy redirects the user. Type the redirect URL in the box provided. Note that in a Redirect ending, you can specify session variables for the URI. For example, you can specify that the redirect use the session logon protocol (**http** or **https**), the session start URI (for example, **www.siterequest.com**) and the session start path (for example, **/owa**).

```
%{session.logon.protocol}://%{session.network.name}/%
{session.start.path}
```

**Figure 5.3** Session variables in a redirect URL

- **Deny**  
Specifies the user is not allowed access to the network access resource, and presents a Denied page. To customize the Denied page, see *Customizing the Deny access policy ending*, on page 5-13.
7. To change the color of the ending for better visual clarity in your access policies, click the color square , select a color, and click **Update**.
  8. Click **Save**.



**Figure 5.4** The edit endings dialog, showing a new ending

### To change an access policy ending

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. Click an access policy ending.  
The Select Ending popup screen opens.
4. On the Select Ending popup screen, select an ending for the branch rule.
5. Click **Save**.


### To set a default access policy ending

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. Click the **Edit Endings** button.  
The Endings popup screen opens.
4. Click the Set Default tab.
5. Select the default access policy ending you want to use, and click **Save**.

## Customizing the Deny access policy ending

The Deny access policy ending provides several customized messages that you can configure for the access policy. These include text messages for the logout screen. You can also configure these messages for different languages that you have defined for the access policy.

### To customize the Deny access policy ending

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the corresponding Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. Click the **Edit Endings** button.  
The Endings popup screen opens.
4. On the Deny ending you want to customize, click the plus sign (  ) next to **Customization**.  
The popup screen displays additional setting options.
5. Customize the text for the logon denied settings by typing the text in the corresponding boxes.

Setting	Description
Language	Specifies the language for which you are configuring Deny messages.
Success Title	This message is not currently used.
Success Message	This message is not currently used.
Thank You Message	Specifies a thank you message displayed for network access users after logout.
Error Title	Specifies the text that indicates that the session could not start.
Error Message	Specifies a more specific error message that follows the error title, which indicates that a problem may have occurred during access policy evaluation.
New Session Text	Specifies the text that precedes the link a user clicks to start a new session.
New Session Link	Specifies the text label for the hypertext link to start a new session, such as <b>click here</b> . This link immediately follows the New Session Text.
Session ID Title	Specifies the text that precedes the session number when an error occurs.

Setting	Description
ACL denied page title	Specifies the title text for a page that appears when access is denied by an ACL.
ACL Denied Page Reject Message	Specifies the text that appears when access to a page or site is denied due to an ACL restriction.
ACL Denied Page Return Link Message	Specifies the link text that the user can click to return to the previous page. This is displayed when a user reaches the ACL denied page.

6. Click **Save**.


## Applying an access policy configuration

To complete the configuration of any access policy, and make the access policy active on the server, click the **Apply Access Policy** link at the top of the screen.


## Configuring macros


A macro is a group of reusable checks. Using the visual policy editor, you configure macros in the same way that you configure access policies. The difference is that you do not configure access policy endings, but instead you configure terminals for a macro.

### To create a macro



1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. Click the **Add New Macro** button.  
The Add New Macro popup screen opens.
4. Select the macro template.  
The macro templates are described in the *Using predefined macro templates*, on page 5-18.
5. In the **Name** box, type a name for the macro.  
This is the name by which the macro appears in the **Add Action** popup screen.
6. Click **Save**.
7. To expand the macro, click the plus sign (  ) next to the macro name.
8. To edit an action, click the action name.  
Edits you make to the actions in a macro are applied to the actions in an access policy, after you add the macrocall to the access policy.
9. Add and remove actions from the macro in the same way you add and remove actions from access policies.
10. When you finish customizing an action, click **Save**.

### To configure macro terminals


1. In the visual policy editor, click the plus sign (  ) next to the macro name to expand the macro for which you want to edit terminals.
2. Click **Edit Terminals**.  
The Edit Terminals popup screen opens.
3. To add a terminal, click **Add Terminal**.
4. Type a name for the terminal.

5. To change the color of the ending for better visual clarity in your access policies, click the Dropper , select a color, and click **Update**.
6. If you want to set a default terminal, click the **Set Default** tab, and select the default terminal.
7. If you want to delete a terminal, click the (x) next to the terminal name.

### To add a macrocall to an access policy

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a branch rule of the access policy, click the plus sign (  ) to add an action.  
The **Add Action** popup screen opens.
4. If **Macrocalls** is not expanded, click the plus sign (  ) next to **Macrocalls**.
5. Select a macro you defined previously and click **Add Item**.  
The macrocall is added to the access policy. You can edit the macro items in the macro definition as required.

### To configure a macro to repeat in an access policy

1. From the visual policy editor, click the plus sign (  ) next to the macro that you want to configure.  
The macro is displayed along with buttons and a list of terminals.
2. Click **Rename/Settings**.  
A popup window opens.
3. From **Maximum Macro Loop Count**, select a number greater than 1 and click **Save**.  
The popup screen closes and **Loop** displays on the list of terminals for the macro.
4. In the macro, click the terminal on the branch that contains the actions that you want to repeat.
5. Select **Loop** and click **Save**.  
The popup screen closes.

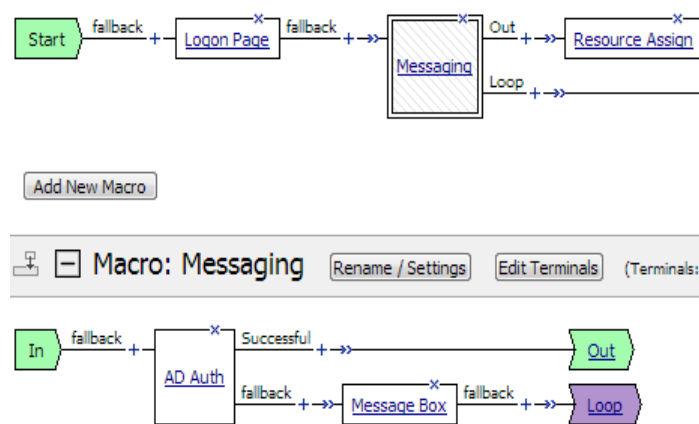


6. After you are done configuring the macro, add the macrocall to the access policy that you are editing.  
A Loop branch follows the macrocall. The access policy takes the Loop branch when a macro exits on the Loop terminal.

## Example macro loop

A user-defined macro, Messaging, contains an authentication action and a message action. See Figure 5.5 for the macro definition and to see the macrocall in an access policy.

**Figure 5.5** Example macro loop



When the macro runs and the authentication action passes, the macro exits on the Out terminal to an Out branch in the access policy. When the authentication action fails, the message action runs and the macro loops back to the authentication action again. If the macro count exceeds the maximum, the macro exits on the Loop terminal to a Loop branch.

## To delete a macro

Click the (x) button at the right of the screen next to the macro name. You can delete a macro only if it is not in use.

## Using predefined macro templates

You can use predefined macro templates to create macros for use in your policies. The following sections describe a few of the available macro templates and are offered as examples.

- *Using the empty macro template*, on page 5-18
- *Using the AD auth and resources macro template*, on page 5-18
- *Using the SecurID and resources macro template*, on page 5-19

---

### ◆ Tip

*If you open these macro definitions to view them, you can better understand how the macros are configured. Each macro definition includes instructions on how to add and open the macro template.*

## Using the empty macro template

You can use the empty macro template to add an unconfigured macro template that includes only a start point and an end point to the access policy. Use this as a starting point to configure a new macro for an access policy.

## Using the AD auth and resources macro template

The AD auth and resources macro template is a preconfigured macro template that adds Active Directory authentication to your access policy.

This macro template includes:


- a start point (In)
- a logon page action
- an Active Directory authentication action
- a resource assign action, that follows a successful Active Directory authentication
- successful and failure terminals

## Configuring the AD Auth and resources macro template

In this macro template, you must configure both the Active Directory action and the resource assign action. You can optionally customize the logon page action with custom messages, and localized messages for different languages.

### To add and customize the AD auth and resources macro

1. In the visual policy editor, click the **Add New Macro** button. The Macro Template popup screen opens.
2. Select the macro template **AD Auth and resources**.

3. Click **Save**.  
The popup screen closes.
4. To expand the macro, click the  (plus) next to the macro name.
5. To edit an action, click the action name.  
In the macro display, the action popup screen opens.
  - To customize the Active Directory action, see the ***BIG-IP® Access Policy Manager® Authentication Configuration Guide***.
  - To customize the resource assign action, see *Assigning resources*, on page 6-8.
  - To customize the logon page action, see *To customize the logon page action*, on page 11-2
6. When you finish customizing an action, click **Save**.
7. To add this macro to the access policy, see *To add a macrocall to an access policy*, on page 5-16.

## Using the SecurID and resources macro template

The SecurID and resources macro template is a preconfigured macro template that adds SecurID authentication to your access policy.


This macro template includes:

- a start point (In)
- a logon page action
- an SecurID authentication action
- a resource assign action, that follows a successful SecurID authentication
- successful and failure terminals

## Configuring the SecurID and resources macro template

In this macro template, you must configure both the SecurID action and the resource assign action. You can optionally customize the logon page action with custom messages, and localized messages for different languages.

### To add and customize the SecurID and resources macro

1. In the visual policy editor, click the **Add New Macro** button.  
The Macro Template popup screen opens.
2. Select the macro template **SecurID and resources**.
3. Click **Save**.  
The popup screen closes.
4. To expand the macro, click the  (plus) next to the macro name.
5. To edit an action, click the action name.  
In the macro display, the action popup screen opens.

- To customize the SecurID action, see the *BIG-IP® Access Policy Manager® Authentication Configuration Guide*.
- To customize the resource assign action, see *Assigning resources*, on page 6-8.
- To customize the logon page action, see *To customize the logon page action*, on page 11-2

6. When you finish customizing an action, click **Save**.

To add this macro to the access policy, see *To add a macrocall to an access policy*, on page 5-16.

## Exporting and importing access profiles

You can export any access profile, and later restore that access profile, or import it to another Access Policy Manager®. Exported profiles are saved as files with the extension **conf**.

When you import a profile, you select a **conf** file. You also specify a name for the new profile, and whether to reuse existing objects, like resources.

### To export an access profile

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. Locate the access profile you want to export. In the Export column, click the **Export** link.  
You are prompted to open or save a **conf** file.
3. Specify a location and save the file.

### To import an access profile

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. Click the **Import** button.  
The Import Profile screen opens.
3. In the **New Profile Name** box, type the name for the new policy.
4. Next to the Config File Upload box, click **Browse**.
5. Select a **conf** file to import and click the **Open** button.
6. Select the **Reuse Existing Objects** check box to reuse objects that exist on the server.  
This option reuses objects that exist on the server, such as server definitions or resources, instead of recreating them for use with this policy.
7. Click **Import**.  
The file is imported to the system.





# 6

---

## Configuring Logon, Assignment, and General Purpose Actions

---

- Configuring actions in an access policy





## Configuring actions in an access policy

In the visual policy editor, you can add and configure general purpose actions to customize your access policy. You can add logon pages, assign resources and variables, select a route domain for policy-based routing, add logging of specific session variables, or add messages and provide decisions in access policies or access policy macros. The action tasks you can do include:

- *Adding and customizing a logon page*, following
- *Adding an HTTP 401 response page*
- *Adding an external logon page*, on page 6-7
- *Assigning resources*, on page 6-8
- *Assigning variables*, on page 6-10
- *Adding a virtual keyboard to the logon screen*, on page 6-13
- *Adding SSO credential mapping*, on page 6-14
- *Filtering access with Citrix SmartAccess filters*, on page 6-15
- *Selecting a route domain or SNAT*, on page 6-16
- *Adding access policy logging*, on page 6-17
- *Adding a message box*, on page 6-17
- *Adding a decision box*, on page 6-18
- *Adding a dynamic ACL*, on page 6-19
- *Adding an iRule event*, on page 6-20

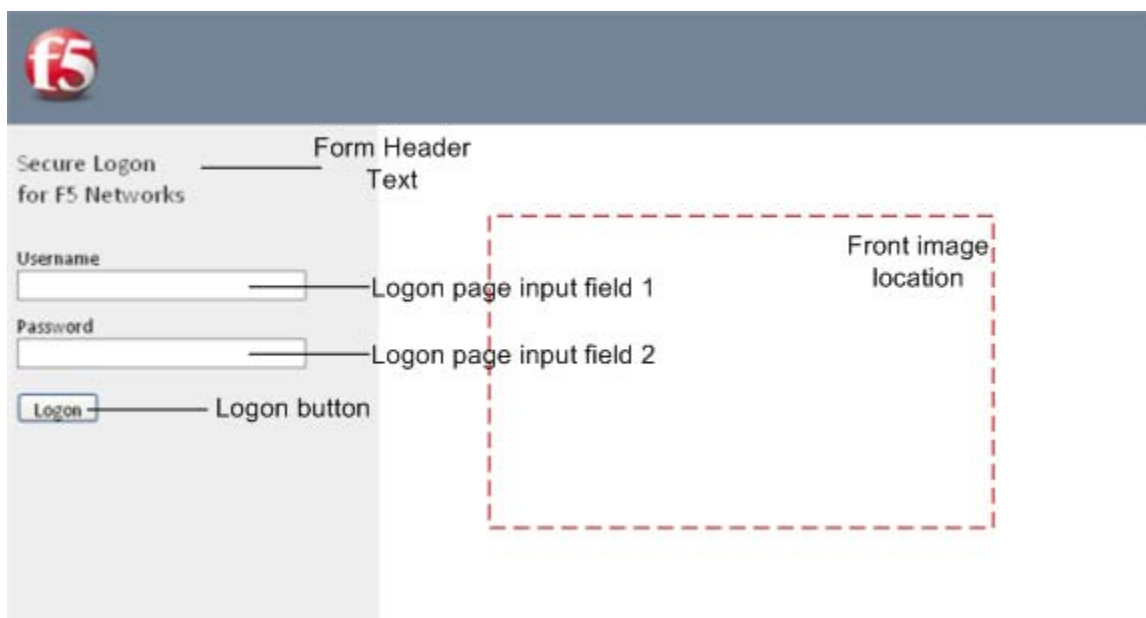
## Adding and customizing a logon page

You can customize the logon page with custom fields and text for different sections of the logon form. On the logon page you can also localize text messages for different languages. The logon page displays up to five logon page agents that can be fully customized. You can define a logon page agent with the following elements and options:

- **CAPTCHA configuration** - Select a configuration from the list to display a CAPTCHA challenge on the logon page. The challenge is displayed after the number of logon failures exceeds a configured limit. A CAPTCHA challenge generates and grades tests, such as the ability to decipher distorted text.
- **Split domain from full username** - Select **Yes** to specify that when a username and domain combination is submitted (for example **marketing\jsmith** or **jsmith@marketing.example.com**), only the username portion (in this example, "**jsmith**") is stored in the session variable **session.logon.last.username**. If you select **No**, the entire username string is stored in the session variable.
- **Type** - Specifies the type of logon page agent. You can specify any agent to be **text**, **password**, or **none**.

- A **text** agent type displays a text field, and shows the text that is typed in that field.
- A **password** agent type displays an input field, but displays the typed text input as asterisks.
- A **none** agent type specifies that the field is not displayed on the logon page.
- **Post Variable Name** - Specifies the variable name that is prepended to the data typed in the text field. For example, the POST variable **username** sends the user name input **omaas** as the POST string **username=omaas**.
- **Session Variable Name** - Specifies the session variable name that the server uses to store the data typed in the text field. For example, the session variable **username** stores the username input **omaas** as the session variable string **session.logon.last.username=omaas**.
- **Read Only** - Specifies whether the logon page agent is read-only, and always used in the logon process as specified. You can use this to add logon POST variables or session variables that you want to submit from the logon page for every session that uses this access policy. You can use a read only logon page field to populate a field with a value from a session variable.  
For example, you can use the On-Demand Certificate agent to extract the **CN** (typically the user name) field from a certificate, then you can assign that variable to **session.logon.last.username**. In the logon page action, you can specify **session.logon.last.username** as the session variable for a read only logon page field that you configure. When Access Policy Manager displays the logon page, this field is populated with the information from the certificate CN field (typically the user name).

Figure 6.1 shows some items that can be customized with the logon page action.



**Figure 6.1** Items that you can customize with the logon page action

### To add and customize a logon page action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen appears.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a rule branch of the access policy, click the plus sign ( **+>** ) to add an action.  
The Add Item popup screen opens.
4. On the Logon tab, select **Logon Page** and click **Add Item**.  
The Logon page action popup screen opens.
5. To present a CAPTCHA challenge on the logon page, select a configuration from the **CAPTCHA configuration** list.
6. In the Logon Page Agent section, enable the fields you want to display on the logon page.  
By default, a text field for user name, and a password field for the password are enabled and displayed. You can specify up to three more fields to display, or customize the ones enabled.
7. From the Language list, select the language for which you want to customize messages.  
The four default languages include English (**en**), Japanese (**ja**),

simplified Chinese (**zh-tw**), and traditional Chinese (**zh-cn**). You can specify more languages in the Access Profile properties **Language Settings** section.

8. Customize the logon page elements:
  - **Form Header Text**  
Specifies the text that appears at the top of the logon box.
  - **Logon Page Input Field # (1-5)** - These fields specify the text that is displayed on the logon page for each of the logon page agents, defined in the Logon Page Agent screen area.
  - **Save Password Checkbox**  
Specifies the text that appears adjacent to the check box that allows users to save their passwords in the logon form. This field is used only in the secure access client, and not in the web client.
  - **Logon Button**  
Specifies the text that appears on the logon button, which a user clicks to post the defined logon agents.
  - **Front Image**  
Specifies an image file to display on the logon page.  
Click **Browse** to select a file from the file system. Click **Show image** or **Hide Image** to show or hide the currently selected image file. Click **Revert to Default Image** to discard any customization and use the default logon page image.
  - **New Password Prompt**  
Specifies the prompt displayed when a new Active Directory password is requested.
  - **Verify Password Prompt**  
Specifies the prompt displayed to confirm the new password when a new Active Directory password is requested.
  - **Password and Password Verification do not Match**  
Specifies the prompt displayed when the new Active Directory password and verification password do not match.

### To create a CAPTCHA configuration

1. On the Main tab of the navigation pane, expand **Access Policy**, click **Access Profiles** and click **CAPTCHA Configurations**.  
The CAPTCHA Configurations List screen displays.
2. Click **Create**.  
The New CAPTCHA Configuration screen displays.
3. In the **General Properties** area:
  - a) In the **Name** field, type a name for the configuration.
  - b) In the **Description** field, type a description.
4. In the **Configuration** area, configure these settings:
  - **Private Key** - Type the string that was provided as the private key when you signed up for CAPTCHA service.

- **Public Key** - Type the string that was provided as the public key when you signed up for CAPTCHA service.
- **Verification URL** - Type the URL of the service that verifies the response to the CAPTCHA challenge. Defaults to `www.google.com/recaptcha/api/verify`. Do not start this URL with **https**.
- **Challenge URL** - Type the URL of the service that provides the CAPTCHA challenge. Defaults to `www.google.com/recaptcha/api/challenge`. Do not start this URL with **https**.
- **Noscript URL** - Type the URL to use for obtaining the challenge picture if JavaScript is disabled. Defaults to `www.google.com/recaptcha/api/noscript`. Do not start this URL with **https**.
- **Display CAPTCHA after** - Type the number of unsuccessful logon attempts to allow before issuing a CAPTCHA challenge. Defaults to 0 (zero), in which case APM always issues a challenge on logon failure.
- **Track Failures** - Choose one or more options to specify how to track logon failure attempts:
  - **IP address** - Checks whether the configured number of unsuccessful logon attempts has been exceeded for this IP address.
  - **Username** - Checks whether the configured number of unsuccessful logon attempts has been exceeded for the provided username.
  - **Theme** - To control the appearance of the CAPTCHA widget, select a standard theme or select Custom. Defaults to Red. When you select Custom, you must do some coding to implement the look and feel that you want for the CAPTCHA challenge; for information, refer to the site you use for CAPTCHA service.

*If you select the Custom theme, but do not add code to the logon page, the CAPTCHA challenge is not displayed. However, the CAPTCHA challenge requires a response. In this case, users cannot respond and cannot log in.*

5. Click **Finished**.

## Adding an HTTP 401 response page

The HTTP 401 response logon page allows you to send an HTTP 401 Authorization Required Response page to capture HTTP Basic or Negotiate authentication in your access policy, and provide branches for Basic and HTTP authentication. You can define the HTTP 401 response page with the following elements and options:

- **Split domain from full username** - Select **Yes** to specify that when a username and domain combination is submitted (for example **marketing\jsmith** or **jsmith@marketing.example.com**), only the username portion (in this example, "**jsmith**") is stored in the session variable **session.logon.last.username**. If you select **No**, the entire username string is stored in the session variable.
- **Basic Auth Realm** - Specify the auth realm for Basic authentication.
- **HTTP Auth Level** - Specify the authentication required for the access policy. You can specify **Basic**, **Negotiate**, **Basic + Negotiate**, or **None**.
- In the Customization section, you can customize the message that appears on the HTTP 401 response page. Note that you can only select languages that are accepted in the access profile, for which you want to customize messages.

## Adding an external logon page

You can add a link to an external logon page to use for logon credentials. This can be used with an external solution to provide robust logon credentials to the access policy.

When the user reaches the external logon page action, the following occurs.

- The Access Policy Manager sends an HTML page containing JavaScript code that redirects users to the external server.
- The client submits a **post\_url** variable. This post variable is used by the external application to return a value to the access policy. When the user completes authentication on the external server, the external server posts back to the URL specified in this variable, to continue the session.

The value of **post\_url** is in the format:

**http(or https)://<Access\_Policy\_Manager\_URI>/my.policy**. The **<Access\_Policy\_Manager\_URI>** is the URI visible to the user, taken from the HTTP Host header value sent by the browser.

## HTML content sample for external logon page submission

Figure 6.2 shows the content of a sample submission to an external logon server from the external logon page action.

```
<html>
  <body>
    <FORM name=external_data_post_cls method=post action="">
      <input type=hidden name=client_data value="SecurityDevice">
      <input type=hidden name=post_url value="https://IP_address_of_virtual/my.policy">
    </FORM>
    <script>
      document.external_data_post_cls.action = unescape("https://external_server_IP_address/loginform2.1.php");
      document.external_data_post_cls.submit();
    </script>
  </body>
</html>
```

*Figure 6.2 External logon page submission sample*

## Sample request from external logon page to virtual server

After the external logon server validates the user, the external server must return the user to the URL specified in **post\_url**, and must post the **username** and **password** variables, which are then used by Access Policy

Manager to validate the user, as shown in Figure 6.3.

```
POST /my.policy HTTP/1.1
Accept: image/gif, image/x-bitmap, image/jpeg, image/pjpeg, application/vnd.ms-excel,
application/vnd.ms-powerpoint, application/msword, application/x-ms-application, application/x-ms-xbap,
application/vnd.ms-xpsdocument, application/xaml+xml, application/x-silverlight, */*
Referer: https://external_server_IP_address/loginform2.1.php
Accept-Language: en,zh-tw;q=0.8,zh-cn;q=0.5,ja;q=0.3
Content-Type: application/x-www-form-urlencoded
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; InfoPath.1; .NET CLR 2.0.50727; .NET CLR
3.0.04506.648; .NET CLR 3.5.21022; .NET CLR 3.0.4506.2152; .NET CLR 3.5.30729)
Host: virtual_server_IP_address
Connection: Keep-Alive
Cache-Control: no-cache
Cookie: LastMRH_Session=733e8a16; MRHSession=254dbb61dcfb45db80e026f3733e8a16
username=1031ntg0x&password=71xu1zjoj
```

**Figure 6.3** External logon page request to Access Policy Manager virtual server

### To add an external logon page action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen appears.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a rule branch of the access policy, click the plus sign ( **+** ) to add an action.  
The Add Item popup screen opens.
4. On the Logon tab, select **External Logon Page** and click **Add Item**.  
The External Logon page action popup screen opens.
5. In the **External Logon Server URI** box, type the external logon page URI.
6. Click **Save** when you are finished.

## Assigning resources

You assign access control lists, a network access resource, portal access resources, a webtop, and webtop links to the access policy using one of the resource assign actions. Each resource assign action provides a similar function, with the following differences.



- **Advanced resource assign** - allows you to assign all resources: network access, portal access, app tunnels, remote desktops, ACLs, webtops, and webtop links
- **Resource assign** - assigns connection resources only: network access, portal access, app tunnels, and remote desktops
- **ACL assign** - assigns static ACLs only
- **Webtop and links assign** - assigns a webtop and webtop links only

Each of these resources contains configuration items. You must assign a network access resource for a network access connection. For portal access, app tunnels, or remote desktops, you must assign the appropriate resources. You can assign a network access resource for a single network access resource, a portal access resource for a portal access resource, or a full webtop to display multiple access types and webtop links. For a web access management connection, you do not assign a connection resource or a webtop. You assign ACLs to all access types with the full resource assign action or with the ACL assign action.

### To add a resource assign action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a rule branch of the access policy, click the plus sign ( **+** ) to add an action.  
The Add Item popup screen opens.
4. On the Assignment tab, select the resource assign action you want to use, and click **Add Item**.  
The resource assign action popup screen for the action you chose opens.
5. For the full resource assign action, click **Add new entry**, then click the **Add/Delete** link. For all other resource assign actions, click the **Add/Delete** link.  
Resource assignment entries appear on the same screen or on a popup screen.
6. To add resources, select the check boxes or click the radio buttons. To remove resources, clear the check boxes or radio buttons.  
For webtops and network access resources, you can only add a single resource with a resource assignment action.
7. Click **Update** if you are using the Advanced resource assign action.
8. Click **Save** to save the action.

## Assigning variables

You use the variable assign action to assign configuration variable, a predefined session variable, or a custom variable resource variable to a AAA server attribute or to a custom expression. This allows you, for example, to assign a custom lease pool for a network access resource, based on the path in an access policy.

After the procedure for how to use the variable assign action, this section includes two simple examples. For an example scenario that uses the variable assign action with a Tcl expression to provide more advanced functionality, see *Using advanced access policy rules*, on page 11-16.

For a list of the configuration variables you can assign with the variable assign action, and the accepted formats for replacement values, see *Understanding network access resource variable attributes*, on page 14-12.

### To add a variable assign action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a rule branch of the access policy, click the plus sign ( **+** ) to add an action.  
The Add Item popup screen opens.
4. On the Assignment tab, select **Variable Assign** and click **Add Item**.  
The Variable Assign action popup screen opens.
5. Click **Add new entry**.
6. Under Assignment, click **change**.  
The Variable Assignment popup screen opens.
7. In the left pane of the Variable Assignment popup screen, select the variable to assign.  
Select **Configuration Variable** to select a variable from a network access or app tunnel resource on the system. Select **Custom Variable** to define a custom variable, and type the custom variable name in the box. Select **Predefined Session Variable** and select the type, name, and property from the current configuration.
8. Select **Secure** to define the session variable as secure.  
A secure session variable is stored in encrypted form in the session database. The secure session variable value is not displayed in the session report, or logged by the logging agent.
9. In the right pane of the Variable Assignment popup screen, select the value to assign the variable.  
You can select **AAA Attribute** and select the RADIUS, LDAP, or

Active Directory agent type, attribute type, and attribute name, or you can select **Custom Expression** and type a custom expression in the box.

10. Click **Finished** when you have assigned the variable.
11. Click **Save** to save the action.

## Example: Overwriting a lease pool with a AAA server attribute

In this example, you assign a lease pool to the network access client by using the custom attribute **myAttribute** from the Microsoft Active Directory server. Access Policy Manager gets the value of myAttribute from the Active Directory server, and replaces the network access resource value for **leasepool\_name** with the value of **myAttribute**. For example, if you assigned **myAttribute** a value of **leasepool1** on the Active Directory server, the network access resource, after the variable assign action, would assign the lease pool **leasepool1** to the user.

### ◆ Note

---

*To use this example, you must have a lease pool defined on the Access Policy Manager, and the name of that lease pool must be defined as the user attribute, **myAttribute**, on the Active Directory server.*

### To overwrite a lease pool with a AAA server attribute

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen appears.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a rule branch of the access policy, click the plus sign ( **+** ) to add an action.  
The Add Item popup screen opens.
4. On the Assignment tab, select **Variable Assign** and click **Add Item**.  
The Variable Assign action popup screen opens.
5. Click **Add new entry**.
6. Under Assignment, next to **empty**, click **change**.  
The Variable Assignment popup screen opens.
7. In the left pane, select **Configuration Variable**.
8. From the **Type** list, select **Network Access**.
9. From the **Name** list, select a network access resource.
10. From the **Property** list, select **leasepool\_name**.
11. In the right pane, select **AAA Attribute**.

12. From the **Agent Type** list, select **AD**.
13. From the **Attribute Type** list, select **Use user's attribute**.
14. In the **AD Attribute Name** box, type **myAttribute**.
15. Click **Finished**.
16. Click **Save** to save the action.

When a user reaches this action in the access policy, Access Policy Manager gets the value for **myAttribute** from the user's AAA attributes, and replaces the lease pool defined in the network access resource with this value.

## Example: Overwriting a lease pool with a custom expression

In this example, you assign a lease pool to the network access client by replacing the network access resource value for **leasepool\_name** with the value of a custom expression. Access Policy Manager evaluates the custom expression, and replaces the network access resource value for **leasepool\_name** with the value of the custom expression. In this example, the access policy replaces the lease pool with an existing lease pool, called **leasepool1**, on the Access Policy Manager. The value you use for the custom expression is a simple string.

### To overwrite a lease pool with a AAA server attribute

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen appears.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a rule branch of the access policy, click the plus sign ( **+** ) to add an action.  
The Add Item popup screen opens.
4. On the Assignment tab, select **Variable Assign** and click **Add Item**.  
The Variable Assign action popup screen opens.
5. Click **Add new entry**.
6. Under Assignment, next to **empty**, click **change**.  
The Variable Assignment popup screen opens.
7. In the left pane, select **Configuration Variable**.
8. From the **Type** list, select **Network Access**.
9. From the **Name** list, select a network access resource.
10. From the **Property** list, select **leasepool\_name**.
11. In the right pane, select **Custom Expression**.

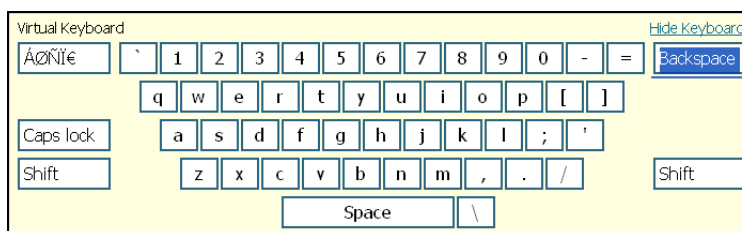
12. In the **Custom Expression** box, type “**leasepool1**” (including the quotes).
13. Click **Finished**.
14. Click **Save** to save the action.

When a user reaches this action in the access policy, Access Policy Manager evaluates the custom expression, in this case, a simple string with the lease pool name, and replaces the lease pool defined in the network access resource with this value.

## Adding a virtual keyboard to the logon screen

You can add a virtual keyboard to the logon screen to prevent password characters from being typed on the physical keyboard. When you add the virtual keyboard action, the virtual keyboard appears on the logon screen when a user clicks in the password field, as shown in Figure 6.4. Users then type the password by clicking the characters on the virtual keyboard, instead of typing them on the physical keyboard.

A virtual keyboard action applies to all logon page actions that follow it in the access policy.



**Figure 6.4** Virtual keyboard on the logon screen

### To add a virtual keyboard action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen appears.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a rule branch of the access policy, click the plus sign ( **+** ) to add an action.  
The Add Item popup screen opens.

***Note:** Add the virtual keyboard in front of a logon page action with which you want to virtual keyboard to be used.*

4. On the General Purpose tab, select **Virtual keyboard** and click **Add Item**.  
The Virtual keyboard action popup screen opens.
5. From the **Virtual Keyboard** list, select **Enabled** to enable the virtual keyboard, or **Disabled** to disable the virtual keyboard.
6. From the **Move Keyboard After Every Keystroke** list, select **Enabled** to move the virtual keyboard after the user clicks each keystroke, or **Disabled** to not move the virtual keyboard after each keystroke.  
This option can further obscure the password that you type with the virtual keyboard.
7. From the **Allow Manual Input** list, select **Enabled** to allow the user to type the password with the physical keyboard or the virtual keyboard. Select **Disabled** to allow the user to type the password only with the virtual keyboard.
8. Click **Save** when the fields are customized.

## Adding SSO credential mapping

You add the SSO credential mapping action to enable users to forward stored user names and passwords to applications and servers automatically, without having to input credentials repeatedly. This allows single sign-on (SSO) functionality for secure access users.

As different applications and resources support different authentication mechanisms, the single sign-on system may be required to store and translate credentials that differ from the user name and password that a user inputs on the logon page. The SSO credential mapping action allows for credentials to be retrieved from the logon page, or in another way for both the user name and the password.

## Understanding SSO token user name caching

The secure access server can cache the user name for use with single sign-on (SSO) applications in the enterprise. When configuring credential caching and mapping, the administrator can define the cached credentials for the SSO Token Username by selecting one of the following:

- **Username from logon page** - Retrieves and caches the user name that is entered on the secure access logon page.
- **sAMAccountName from Active Directory** - Looks up the user's value for sAMAccountName in Active Directory, retrieves the value, and caches it for use as the user name.

- **sAMAccountName from LDAP Directory** - Looks up the user's value for sAMAccountName in the LDAP Directory, retrieves the value, and caches it for use as the user name. This can only be used when the session is configured to access Active Directory over LDAP.
- **Custom** - Allows you to retrieve a custom value from a session variable.

## Understanding SSO token password caching

The secure access server can cache the password for use with single sign-on applications in the enterprise. When configuring credential caching and mapping, the administrator can define the cached credentials for the SSO Token Password by selecting one of the following:

- **Password from logon page** - Retrieves and caches the password that is entered on the secure access logon page.
- **Custom** - Allows you to retrieve a custom value from a session variable.

For information on how to configure SSO with credential caching and proxying, refer to the *BIG-IP® Access Policy Manager® Single Sign-On Configuration Guide*.

## Filtering access with Citrix SmartAccess filters

Use Citrix SmartAccess filters to enable the access policy to act as the Citrix Web Interface, and send SmartAccess filters to the XenApp server, which then displays applications and applies policies based on the filter content.

## Configuring Citrix SmartAccess

Citrix SmartAccess uses the Citrix SmartAccess agent in the access policy to provide filters.

For SmartAccess to work with Access Policy Manager, the **Farm Name** for the filter on the Citrix server must be set to **APM**.

### To add a Citrix SmartAccess filter

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a rule branch of the access policy, click the plus sign ( **+** ) to add an action.  
The Add Item popup screen opens.
4. On the Assignment tab, select **Citrix SmartAccess** and click **Add Item**.  
The Citrix Smart Access action popup screen opens.

5. In the **Assignment** field, type a Citrix SmartAccess filter name.  
For example:  
**Filter2**
6. To add another SmartAccess filter, click **Add new entry**.
7. When you have finished, click **Save** to save the action.

## Selecting a route domain or SNAT

You select a route domain to use route domain-based policy routing. Add this action on a branch of the access policy when you want to send the user to a different route domain, based on the outcomes of previous branches in the access policy. You can select a SNAT to provide Secure NAT to the self IP address of the BIG-IP device, or to choose from a pool of configured internal addresses for SNAT.

SNAT precedence is determined according to the following rules:

- First, if a SNAT is defined in a Network Access resource configuration, that SNAT is used.
- If there is no SNAT defined in the Network Access resource, or the resource is another type, the SNAT is taken from this assignment in the access policy.
- If there is no SNAT assigned in the access policy, the SNAT from the virtual server definition is used.

### To add a route domain selection action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a rule branch of the access policy, click the plus sign ( **+** ) to add an action.  
The Add Item popup screen opens.
4. On the Assignment tab, select **Route Domain and SNAT Selection** and click **Add Item**.  
The Route Domain Selection action popup screen opens.
5. From the **Route Domain ID** list, select a route domain ID to use with this access policy.
6. From the **SNAT** list, select a SNAT pool, **automap**, or **none**.  
Route domains and SNAT pools must be already defined on the Access Policy Manager. For more information, see *Configuring policy routing*, on page 11-10.




## Adding access policy logging

Use access policy logging to write the values of specific session variables or session variable categories to the system logs. You can use this action to trace the session variables that are created for a specific category, or in a specific branch.

One use for access policy logging is to trace the variables created from AAA server attributes. The Access Policy Manager creates session variables for all AAA server attributes, so the session variables that are created in a session are specific to the configuration of the AAA server. As an example, to determine the session variables created from RADIUS attributes, you can set the logging action to log all RADIUS variables, by selecting RADIUS from the **Session Variables** category list.

### To add a logging action


1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a rule branch of the access policy, click the plus sign (  ) to add an action.  
The Add Item popup screen opens.
4. On the General Purpose tab, select **Logging** and click **Add Item**.  
The logging action popup screen opens.
5. Click **Add new entry**.
6. Select a category of session variables to write to the log.
  - If you select a predefined category, all session variables for that session variable category are logged using wildcards. For example, for Active Directory, the session variables **session.ad.last.\*** are logged.
  - If you select the **Custom** category, you can type a session variable or session variable category to log in the **Session Variables** box.
7. To log more session variables, or session variable categories, click **Add new entry**.
8. When you have finished, click **Save** to save the action.

## Adding a message box

You can add a message box anywhere in an access policy. A message box has no effect on the user's access to the network or the access policy checks. It is used solely to present a message to the user, and to prompt the user to

click a link to continue. You might use a message box to warn a user that he is going to a quarantine network, or that the client certificate failed to authenticate, or any other time you want to tell the user a message about the results of a rule branch in the access policy.

### To add a message


1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a rule branch of the access policy, click the plus sign (  ) to add an action.  
The Add Item popup screen opens.
4. On the General Purpose tab, select **Message Box** and click **Add Item**.  
The Message Box action popup screen opens.
5. From the **Language** list, select the language for the message.
6. In the **Message** box, type the message to the user. You can use HTML tags for formatting, as in the example:  
**<font color=red> Please click the link below to continue. </font>**
7. In the **Link** box, type the text that the user must click to continue.  
This text appears as a link the user can click to continue.
8. Click **Save**.

## Adding a decision box

You can add a decision box anywhere in an access policy. You use a decision box to present two options to the user. These options are presented as link text, preceded by images. You might use a decision box when a user fails an endpoint security check, or when a user fails to authenticate. In these cases, one branch can provide an option to allow the user to continue onto a quarantine network that provides only limited access to a segregated subnet. The other branch can provide an option to log out, and present the user with a logon denied ending. Another use of the second option branch is to allow the user to continue to a redirect ending that takes the user to a helpful URL, for example, to the web site of an antivirus vendor to download virus database updates.

### To add a decision box action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.


2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a rule branch of the access policy, click the plus sign (  ) to add an action.  
The Add Item popup screen opens.
4. Select the language to customize for the decision box.
5. In the **Message** box, type a message to the user. You can use HTML tags for formatting, as in the example:  
**<font color=red> Please choose one of the following two options below. </font>**
6. From the **Field 1 image** list, select the image for field one.  
This image precedes the text you type in the next step.
7. In the **Option 1** box, type the text for option 1.  
This text appears to the user as the first clickable link.
8. From the **Field 2 image** list, select the image to use for option 2.  
Note that option 2 is the fallback rule branch of the access policy action. This image precedes the text you type in the next step.
9. In the **Option 2** box, type the text for option 2.  
Note that option 2 is the fallback rule branch of the access policy action. This text appears to the user as the second clickable link.
10. Click **Save**.

## Adding a dynamic ACL

You can add a dynamic ACL after an authentication that captures attributes from the AD, LDAP, or RADIUS attribute, and before the resources are assigned. To add a dynamic ACL, you must complete several steps first.

See *Configuring dynamic ACLs*, on page 3-8 for more information.

### To assign a dynamic access control list with the Dynamic ACL action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a rule branch of the access policy, click the plus sign (  ) to add an action.  
The Add Item popup screen opens.

4. On the Assignment tab, select **Dynamic ACL**, and click **Add Item**. The Dynamic ACL action popup screen opens.
5. To add one or more ACLs, click the **Add new entry** button.
6. To use an F5 ACL from an AD, RADIUS, or LDAP directory, select **Custom**. To use a Cisco AV-Pair ACL from a RADIUS directory, select **Cisco AV-Pair VSA**.
7. In the **Source** field, type the attribute from which the Dynamic ACL action extracts ACLs.  
  
If you are using Cisco AV-Pair VSA from a RADIUS server, the field is prepopulated with **session.radius.last.attr.vendor-specific.1.9.1**.
8. From the **ACL** list, select the dynamic ACL container.
9. From the **Format** list, select the format in which the ACL is specified.
10. To add another ACL entry, click the **Add new entry** button and repeat the procedure.
11. Click **Save** to save the action.

The dynamic ACL action appears in the access policy.

## Adding an iRule event

You can add an iRule event anywhere in an access policy. You use an iRule event to add iRule processing to an access policy at a specific point.

For a list of supported iRule events, see Appendix 15, *Using Access iRule Events*.

---

### ◆ Note

*iRule event access policy items must be processed and completed before the access policy can continue.*

### To add an iRule event action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a rule branch of the access policy, click the plus sign ( **+** ) to add an action.  
The Add Item popup screen opens.

4. Select **iRule event** and click **Add Item**.  
The Custom iRule Event Agent popup screen opens.
5. In the **ID** box, type the iRule event you want to insert.
6. Click **Save**.





# 7

---

## Configuring Endpoint Security (Client-Side)

---

- Understanding endpoint security (client-side) checks
- Checking for a file
- Checking a machine certificate
- Verifying Windows information
- Checking machine information
- Checking processes
- Setting up Windows registry check
- Setting up Windows cache and session control
- Setting up Windows protected workspace
- Assigning a Windows group policy template
- Checking software on an endpoint





## Understanding endpoint security (client-side) checks

In BIG-IP® Access Policy Manager® access policies, you use *client-side checks* to collect and verify system information. In the visual policy editor, you can use the information collected by client-side checks in an access policy, to enforce a specific security level before granting access to network resources. You can also use this information to perform remediation and protect your network resources. The Access Policy Manager provides these checks as a set of access policy actions that you can use to construct an access policy to evaluate client systems.

Access Policy Manager uses ActiveX controls or browser plug-ins to collect information about client systems. For those clients that do not support browser add-ons or that do not allow browser software installation, the client-side security process can inspect HTTP headers to gather information on the client operating system, including the client operating system and browser type. You can check that a client supports client-side checks with the client-side check capability action. If a client does not support client-side checks, that client can follow a different access policy branch.

While Access Policy Manager provides checks for many client devices, some client-side checks may not be supported on all supported operating systems.

For a complete list and brief descriptions of endpoint security (client-side) checks, see *Understanding available actions*, on page 4-3. The basic process for adding an action to an access policy is the same for each action. Properties and branch rules differ, however, from action to action.

## Checking for a file

You use the file check action for Windows, Macintosh, or Linux to verify the presence of one or more files on a client system. On all supported platforms, the file check action can verify one or more file properties, including the file name, size, date, and MD5 checksum. In addition, the Windows version of the file check action can verify version and signer information.

If a file with the described properties exists, the client is passed to the successful branch. If the file does not exist, or a file exists but one or more properties are not correct, the client is passed to the fallback branch.

## Checking for a file with the file check access policy item

Add a file check action to an access policy in a situation where verifying the presence of a certain file can increase confidence in the security of the client system.

### To add a file check action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a branch of the access policy, click the plus sign ( **+** ) to add an action.  
The Add Item popup screen opens.
4. On the Endpoint Security (Client-Side) tab, select the file check action for your platform:
  - For Windows, select **Windows File** and click **Add Item** to add the action to the access policy.
  - For Macintosh, select **Mac File** and click **Add Item** to add the action to the access policy.
  - For Linux, select **Linux File** and click **Add Item** to add the action to the access policy.  
The File action popup screen opens.
5. Click **Add new entry** to add a file entry to the action.
6. Configure the entry.
  - a) In the **FileName** box, type the name for the file you want to check.  
Note that this is the only setting that is required.

- b) If you want to verify that the MD5 checksum matches, in the **MD5** box, type or paste the MD5 checksum.
  - c) If you require an exact size for the file, in the **Size** box, type the size in bytes.  
 Note that if you type a **0** in this box, no file size check occurs. To check for a 0-byte file, you must instead type the MD5 checksum in the **MD5** box. The MD5 checksum for a 0-byte file is always **d41d8cd98f00b204e9800998ecf8427e**.
  - d) If you want to specify the file creation date, in the **Date** box, type the file creation date. The default date of **1970-01-01 00:00:00** is the same as specifying no date.  
 You can determine the file creation date by right-clicking the file in Windows, and selecting **Properties**. The file creation date must be translated to a 24-hour clock, if your system is not on 24-hour time. For example, you would type the file creation date **Wednesday, February 27, 2008, 1:23:37 PM** in this box as **2008-02-27 13:23:37**. The file creation date is set in UTC, or Greenwich Mean Time (GMT), so the server and client timezones are not the same as the file time, and you must adjust the file time you specify accordingly.
  - e) For Windows file check only, if you require that the file be signed, in the **Signer** box, type the signer.
  - f) For Windows file check only, in the **Version** box, type the version of the file, if you want to specify a version, or greater than or less than a version of the file.
  - g) For Windows file check only, from the **Version Comparison** list, select the version comparison operator. Select **=** if you want the file to be the exact version you specify, select **<** if you want the checked file version to be greater than the version number you specify, and select **>** if you want the checked file version to be less than the version number you specify.
7. To add another file to the action, repeat steps 6-7.
  8. Click **Save** to complete the configuration.

## Example: Using file check

In this example, the administrator adds a Windows file check action, with the requirement that a system file, **wininet.dll**, be present on the client system. The file must be version **6.0.2900.2904**, be **658,432 bytes** in size, and have an MD5 checksum of **38ab7a56f566d9aad31812494944824**.

---

### ◆ Note

*This is not a complete example. For the example to work, you must assign an Allow ending to successful branches. You can assign a network access resource, portal access resources, app tunnels, remote desktops, and a webtop. For a web access management connection, you need not assign resources. This example is configured starting with an empty access policy.*

### To configure the example action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a branch of the access policy, click the plus sign ( **+** ) to add an action.  
The Add Item popup screen opens.
4. On the Endpoint Security (Client-Side) tab, select **Windows File** and click **Add Item** to add the action to the access policy.  
The File action popup screen opens.
5. Click **Add new entry** to add a file entry to the action.
6. Configure the entry:
  - In the **File Name** box, type **C:\Windows\System32\wininet.dll**.
  - In the **MD5** box, type the MD5 checksum **a4f6142caba8d9aad31812494944824**.  
Many MD5 checksum utilities include a copy function to simplify this step.
  - In the **Size** box, type **1392128**.
  - In the **Version** box, type **9.0.8112.2904**.
  - From the **Version Comparison** list, select **=**.

The configured action appears as shown in Figure 7.1.
7. Click **Save** to complete the configuration.

Properties\* [Branch Rules](#)

Name:

---

**Windows File Checker**

Continuously check the result and end the session if it changes

Insert Before:

---

	FileName	MD5	Size (bytes)	Signer	Date	Version	Version Comparison	
1	<input type="text" value="C:\Windows\System32\wininet"/>	<input type="text" value="a4f6142caba8"/>	<input type="text" value="1392128"/>	<input type="text" value=""/>	<input type="text" value="1970-01-01"/>	<input type="text" value="9.0.8112"/>	<input type="text" value="="/>	<input type="button" value="X"/>

**Figure 7.1** Windows file check action example

## Checking a machine certificate

You use the machine certificate authentication check action to check for the presence of a machine certificate on the client computer. You can configure the action to check for a certificate in a specific location, and to require matches with particular certificate fields to pass.

### About checking a machine certificate on a Windows client

On a Windows client, the Machine Cert Auth action accesses the machine certificate private key; admin privilege is required to do this. A user that runs without admin privilege cannot successfully run this check unless the machine certificate checker service is installed on the machine.

To install the machine certificate checker service, you must customize the Windows client package to include the service and download and run the customized client package on the machine. (You can customize and download the Windows client package from the Secure Connectivity area of Access Policy Manager.)

### About checking a machine certificate on a Mac client

On a Mac client, the Machine Cert Auth action accesses the machine certificate private key. If the certificate is stored in a keychain other than user's own keychain, such as the system keychain, then you must set an ACL on the Mac for non-admin users to be able to access this private key.

## Understanding machine cert auth check options

The machine cert auth check can be configured with a number of options. These options are listed below:

- **Certificate Store Name**  
Specifies the certificate store name that the action attempts to match. The certificate store can be a system store with a predefined name like **MY**, or a user-defined name. The store name can contain alphanumeric characters. The default store name is **MY**.  
For a Mac client, if you do not want to use the default location, you must type the name of a keychain file. Type only the file name, which is case-sensitive, without a file path. To view keychains, use the **security list-keychains** command from the terminal.
- **Certificate Store Location**  
Specifies the type and location of the store that contains the certificate, either the local machine or the current user.  
For a Windows client, the store locations are in the following registry locations:
  - **LocalMachine** - searches in **HKEY\_LOCAL\_MACHINE** for the machine certificate.

- **CurrentUser** - searches in **HKEY\_CURRENT\_USER** for the machine certificate.

For a Mac client, the store locations are keychains in the following domains:

- **LocalMachine** – Searches for the machine certificate in the keychain specified in Certificate Store Name in the system available preference domain.
- **CurrentUser** - Searches for the machine certificate available in the keychain specified in Certificate Store Name in the user preference domain.

Here are some examples for Mac clients:

- If **Certificate Store Name = System.keychain** and **Certificate Store Location = LocalMachine**, APM searches for the machine certificate in `/Library/Keychain/System.keychain`.
- If **Certificate Store Name= login.keychain** and **Certificate Store Location = CurrentUser**, APM searches for the machine certificate in `/Users/<username>/Library/Keychains/login.keychain`.
- If **Certificate available Store Name=MY**, APM searches for the machine certificate in the default available keychain of the Certificate Store Location.
- **CA Profile**  
Specifies the certificate authority profile for the machine certificate. To configure a certificate authority, on the navigation pane, expand **Local Traffic**, click **Profiles**, from the SSL menu select **Certificate Authority**, and click **Create**.
- **OCSP Responder**  
Specifies the Online Certificate Status Protocol responder configured to provide certificate status. The OCSP responder is used to check the status of the machine certificate configured in the machine cert auth check action.
- **Certificate Match Rule**  
Specifies how the machine cert auth check action identifies the certificate. The following match rules are supported:
  - **SubjectCN Match FQDN** - Specifies that the common name in the machine certificate matches the computer's fully qualified domain name (FQDN).
  - **SubjectAltName Match FQDN** - Specifies that the content extracted from the **Subject Alternative Name** field, using a specified regular expression, must match the computer's FQDN.  
When this option is selected, the **SubjectAltName** box appears. This box is required for the **SubjectAltName** match value only. The regular expression is used to extract content from the first subgroup matched in the **Subject Alternative Name**, and then to compare the extracted content with the machine's FQDN.

Note that the order of RDNs is the same as is displayed; the required separator is a comma ( , ). Subcases for regex extraction follow:

**Partial extraction.** For example,

`".*DNS Name=([ ^, ]+).*"`

or

`".*Other Name:Principal Name=([ ^, ]+).*"`

For a regular expression

`'.*DNS Name=([ ^, ]+).*'`, the value of the DNS Name field is extracted for matching.

**Whole extraction.** Leave this field empty or use `"(.*)"`, in order to allow the entire SubjectAltName content to be extracted for matching.

- **Any** - Specifies that the first certificate in the specified certificate store is sent to the server for further validation. Any other certificates are ignored.
- **Issuer** - Specifies that the content from the **Issuer** field matches the pattern specified by the regular expression.  
When this option is selected, the **Issuer** box appears. This box is required for the **Issuer** match, as well as **Issuer and Serial Number** match. The regular expression is used to match the **Issuer**'s content against the specified pattern.

Note that the order of RDNs is the same as is displayed; the required separator is a comma ( , ).

Subcases for the regex match are as follows:

**Partial match.** For example,

`"CN=.*, OU=FP, O=F5, L=San Jose, S=CA, C=US"`

**Exact Match.** For example,

`"CN=Root, OU=FP, O=F5, L=San Jose, S=CA, C=US"`

- **Issuer and Serial Number** - Specifies that the content from the **Issuer** field matches the pattern specified by the regular expression, and that the serial number precisely matches your input.  
When this option is selected, the **Issuer** box appears. This box is required for the **Issuer** match, as well as **Issuer and Serial Number** match. The regular expression is used to match the **Issuer**'s content against the specified pattern.  
When this option is selected, the **Serial Number** box appears. The serial number must be an exact match (for example, the hex string must be typed in the same order as it is displayed by OpenSSL and Windows cert tools). For example,  
`0102030405060708090a`.
- **Save Certificate in a session variable**  
Select **Enabled** to save the complete encrypted text of the machine certificate in a session variable,  
`session.windows_check_machinecert.<name>.cert`.




- **Allow User Account Control right elevation prompts**  
Set this option to **No** to suppress the UAC prompt during private key checking for non-admin users.

## Checking a machine certificate with the machine cert access policy item

Use the machine cert auth check action to check for the existence of fields in a machine cert, to ensure that client systems comply with your security policy.

### To add a machine cert auth check action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a branch of the access policy, click the plus sign (  ) to add an action.  
The Add Item popup screen opens.
4. On the Endpoint Security (Client-Side) tab, select **Machine Cert Auth** and click **Add Item** to add the action to the access policy.  
The Machine Cert Auth action popup screen opens.
5. In the **Certificate Store Name** box, type the certificate store name, or use the provided value, **MY**.
6. From the **Certificate Store Location** list, select the certificate store registry location.
7. From the **CA Profile** list, select the certificate authority.
8. From the **OCSP Responder** list, select an OCSP responder, if required, or **None**.
9. From the Certificate Match Rule list, select the desired certificate match rule, and enter values in any related boxes that appear.  
*See [Checking a machine certificate with the machine cert access policy item](#), on page 7-9, for more information.*
10. From the **Save Certificate in a session variable** list, select **Enabled** to save the certificate in a session variable, or **Disabled** to not save the certificate as a session variable.
11. Click **Save** to complete the configuration.

## Example: Using machine cert auth check

In this example, the machine certificate checks the fully qualified domain name for **www.siterequest.com** against the **Subject Alternative Name** field.

### ◆ Note

---

*This is not a complete example. For the example to work, you must assign an Allow ending to successful branches. You can assign a network access resource, portal access resources, app tunnels, remote desktops, and a webtop. For a web access management connection, you need not assign resources. This example is configured starting with an empty access policy.*

### To configure the example action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a branch of the access policy, click the plus sign ( **+** ) to add an action.  
The Add Item popup screen opens.
4. On the Endpoint Security (Client-Side) tab, select **Machine Cert Auth** and click **Add Item** to add the action to the access policy.  
The Machine Cert Auth action popup screen opens.
5. From the **Certificate Match Rule** list, select **SubjectAltName match FQDN**.
6. In the **Subject Alternative Name** box, type **\*.siterequest.com**.
7. Leave all other settings at their default values.
8. Click **Save** to complete the configuration.

## Verifying Windows information

You use the Windows info check action to verify the presence of Windows operating system versions, Windows patches, or Windows updates.

## Setting up Windows info action

Use the Windows info action to determine if the user is using the correct version of Windows, has applied specific patches or updates to Windows, or meets other Windows requirements.

### To add a Windows info action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a branch of the access policy, click the plus sign ( **+** ) to add an action.  
The Add Item popup screen opens.
4. On the Endpoint Security (Client-Side) tab, select **Windows Info** and click **Add Item** to add the action to the access policy.  
The Windows Info action popup screen opens.
5. Click the Branch Rules tab.
6. Click the **Add Branch Rule** button.
7. In the **Name** box, type a name for the rule.
8. Next to **Expression: Empty**, click **change**.  
The Add Expression popup screen opens.
9. Click the **Add Expression** button.
10. From the **Agent Sel** list, select **Windows Info**.
11. From the **Condition** list, select **Windows platform** or **Windows update**.
  - If you selected Windows platform, from the **Windows Platform is** list, select the Windows version.
  - If you selected Windows update, in the **Windows patch** box, type the update name. The format for this can be a KB patch or a Windows service pack, for example **KB869074** or **SP2**.
12. Click **Save** to complete the configuration.

## Example: Using Windows info check

For this example, you add a Windows info check action that contains rules that check for Windows XP and Service Pack 2.

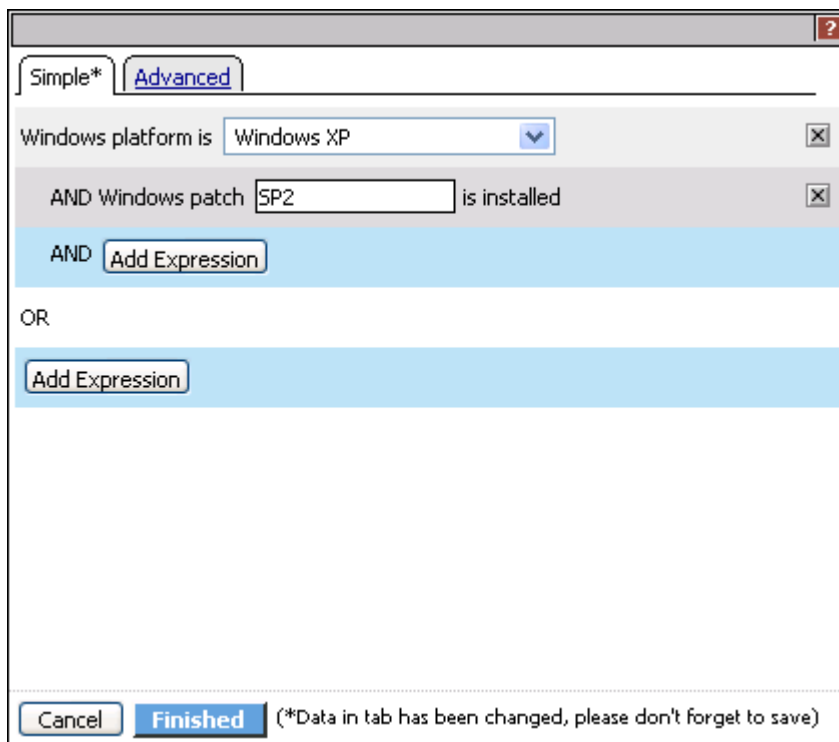
### ◆ Note

*This is not a complete example. For the example to work, you must assign an Allow ending to successful branches. You can assign a network access resource, portal access resources, app tunnels, remote desktops, and a webtop. For a web access management connection, you need not assign resources. This example is configured starting with an empty access policy.*

### To add the example action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a branch of the access policy, click the plus sign ( **+** ) to add an action.  
The Add Item popup screen opens.
4. On the Endpoint Security (Client-Side) tab, select **Windows Info** and click **Add Item** to add the action to the access policy.  
The Windows Info action popup screen opens.
5. Click the Branch Rules tab.
6. Click **Add Branch Rule**.
7. Type the name **XP SP2** for the rule.
8. Next to **Expression: Empty**, click **change**.  
The Expression popup screen opens.
9. Click the **Add Expression** button.  
The popup screen displays new information.
10. From the **Agent Sel** list, select **Windows Info**.
11. From the **Condition** list, select **Windows platform**.
12. From the **Windows Platform is** list, select **Windows XP**.
13. Click the **Add Expression** button.
14. To add the next expression, next to **AND**, click **Add Expression**.  
The popup screen displays new information.
15. From the **Agent Sel** list, select **Windows Info**.
16. From the **Condition** list, select **Windows update**.
17. From the **Windows Platform is** list, select **Windows XP**.

18. In the **Windows Patch** box, type **SP2**.
19. Click the **Add Expression** button.  
The Expression popup screen shows the expression configured as shown in Figure 7.2.  
  
To view the rule you have created, click the Advanced tab. You see the expression  
**expr { [mcget {session.windows\_info\_os.last.platform}] == "WinXP" && [mcget {session.windows\_info\_os.last.updates}] contains "SP2" }**
20. Click **Finished**.
21. Click **Save** to complete the configuration.



*Figure 7.2 Windows information action expression example*

## Checking machine information

Use this action to collect machine info from the client system.

The Machine Info check collects the following information, and creates session variables with it. You can then detect for these session variables using a session variable, or by configuring an expression with the expression builder pull-down menu item Machine Info. Note that in the session variable value, any special characters are represented by ASCII characters. For example, a space character is represented by the value %20. Leading and trailing white space characters are removed.

- **CPU Name** - collects the CPU name from the client system and stores it in the session variable **session.machine\_info.cpu.name**.

Example return value: **Intel(R) Core(TM)2 CPU 6300 @ 1.86GHz**

- **CPU Vendor ID** - collects the CPU vendor from the client system and stores it in the session variable **session.machine\_info.cpu.vendor**.

Example return value: **GenuineIntel**

- **CPU Description** - collects the CPU name from the client system and stores it in the session variable **session.machine\_info.cpu.description**.

Example return value: **x86 Family 6 Model 15 Stepping 2**

- **CPU maximum clock** - collects the CPU maximum clock speed from the client system and stores it in the session variable **session.machine\_info.cpu.max\_clock**.

Example return value: **1860**

- **Motherboard manufacturer** - collects the motherboard manufacturer from the client system and stores it in the session variable **session.machine\_info.motherboard.manufacturer**.

Example return value: **Dell Inc.**

- **Motherboard serial number** - collects the motherboard serial number from the client system and stores it in the session variable **session.machine\_info.motherboard.sn**.

Example return value: **CN156407A704NP**

- **Motherboard product** - collects the motherboard product name from the client system and stores it in the session variable **session.machine\_info.motherboard.product**.

Example return value: **0TY565**

- **BIOS manufacturer** - collects the BIOS manufacturer from the client system and stores it in the session variable **session.machine\_info.bios.manufacturer**.

Example return value: **Dell Inc.**


- **BIOS serial number** - collects the BIOS serial number from the client system and stores it in the session variable **session.machine\_info.bios.sn**.

Example return value: **770QSC1**

- **BIOS version** - collects the BIOS version from the client system and stores it in the session variable **session.machine\_info.bios.version**.  
Example return value: **DELL - 14**
- **Number of network adapters** - collects the number of network adapters from the client system and stores it in the session variable **session.machine\_info.net\_adapter.count**.  
Example return value: **1**
- **(First/Second) network adapter name** - collects the name of the first or second network adapter name from the client system and stores it in the session variable **session.machine\_info.net\_adapter.list.[number].name**. For example, the variable **session.machine\_info.net\_adapter.list.[0].name** retrieves the name of the first network adapter on the system.  
Example return value: **Broadcom NetXtreme 57xx Gigabit Controller**
- **(First/Second) network adapter MAC address** - collects the MAC address of the first or second network adapter from the client system and stores it in the session variable **session.machine\_info.net\_adapter.list.[number].mac\_address**.  
For example, the variable **session.machine\_info.net\_adapter.list.[0].mac\_address** retrieves the MAC address of the first network adapter on the system.  
Example return value: **00:AA:11:BB:33:FF**
- **Number of hard drives** - collects the number of hard drives from the client system and stores it in the session variable **session.machine\_info.hdd.count**.  
Example return value: **2**
- **(First/Second) hard drive model number** - collects the model of the first or second hard drive on the client system and stores it in the session variable **session.machine\_info.hdd.list.[number].model**. For example, the variable **session.machine\_info.hdd.list.[1].model** retrieves the model of the second hard drive on the system.  
Example return value: **ST3160812AS**
- **(First/Second) hard drive serial number** - collects the hard drive serial number from the first or second hard drive on the client system and stores it in the session variable **session.machine\_info.hdd.list.[number].sn**. For example, the variable **session.machine\_info.hdd.list.[0].sn** retrieves the serial number of the second hard drive on the system.  
Example return value: **5LSDN69C**

### Setting up the machine info access policy action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.

2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a branch of the access policy, click the plus sign (  ) to add an action.  
The Add Item popup screen opens.
4. On the Endpoint Security (Client-Side) tab, select **Machine Info** and click **Add Item** to add the action to the access policy.  
The Machine Info action popup screen opens.
5. Click the Branch Rules tab.
6. In the **Name** field, type a name for the **Branch Rule**.
7. Next to **Expression: Empty**, click **change**.  
The Expression popup screen opens.
8. Click the **Add Expression** button.  
The popup screen displays new information.
9. From the **Agent Sel** list, select **Machine Info**.
10. From the **Condition** list, select the condition.
11. In the field that appears type the value to check for the machine info condition.
12. Click the **Finished** button.
13. On the Branch Rules screen that appears, click **Save**.

## Example: Using machine info check

For this example, you add a machine info check action that contains a branch rule that checks for a machine with an Intel processor and a Broadcom NetXtreme 57xx Gigabit Controller.

### **Note**


---

*This is not a complete example. For the example to work, you must assign an Allow ending to successful branches. You can assign a network access resource, portal access resources, app tunnels, remote desktops, and a webtop. For a web access management connection, you need not assign resources. This example is configured starting with an empty access policy.*

### **To add the example action**

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.



2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a branch of the access policy, click the plus sign (  ) to add an action.  
The Add Item popup screen opens.
4. On the Endpoint Security (Client-Side) tab, select **Window Machine Info** and click **Add Item** to add the action to the access policy.  
The Machine Info action popup screen opens.
5. Click the Branch Rules tab.
6. Click **Add Branch Rule**.
7. Type the name **Machine check** for the rule.
8. Next to **Expression: Empty**, click **change**.  
The Expression popup screen opens.
9. Click the **Add Expression** button.  
The popup screen displays new information.
10. From the **Agent Sel** list, select **Machine Info**.
11. From the **Condition** list, select **CPU Vendor ID**.
12. In the **CPU Vendor ID** field, type **GenuineIntel**.
13. Click the **Add Expression** button.
14. To add the next expression, next to **AND**, click **Add Expression**.  
The popup screen displays new information.
15. From the **Agent Sel** list, select **Machine Info**.
16. From the **Condition** list, select **First network adapter name**.
17. In the **First network adapter name** field, type **Broadcom NetXtreme 57xx Gigabit Controller**.
18. Click **Add Expression**.
19. Click **Finished**.
20. Click **Save** to complete the configuration.

Properties Branch Rules\*

Add Branch Rule

Name: Machine info

Expression: CPU vendor ID is GenuineIntel  
OR First network adapter name is Broadcom NetXtreme 57xx Gigabit Controller

Name: fallback

**Figure 7.3** Machine info action expression example

## Checking processes

With the process check action, you can verify that one or more particular processes are or are not running.

You use the process check action with a Boolean expression to check for processes that are running on the client system.

The Boolean expressions you specify can use the wildcards \* and ?, parentheses ( ) to combine values, and the logical operators **AND**, **OR**, and **NOT**.

## Setting up the process check access policy item

You can add process checks for Windows, Linux, or Mac clients.

### To add a process check action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a branch of the access policy, click the plus sign ( **+** ) to add an action.  
The Add Item popup screen opens.
4. On the Endpoint Security (Client-Side) tab, select the **Process Check** for the operating system you are checking, and click **Add Item** to add the action to the access policy.  
The Process Check action popup screen opens.
5. In the **Expression** box, type the expression.
6. Click **Save** to complete the configuration.

## Example: Using process check

In this example, you use the process check action to determine the presence of the running Windows processes **winlogon.exe** and **GoogleDesktop.exe**. You also determine that no process with **gator** in the name is running.

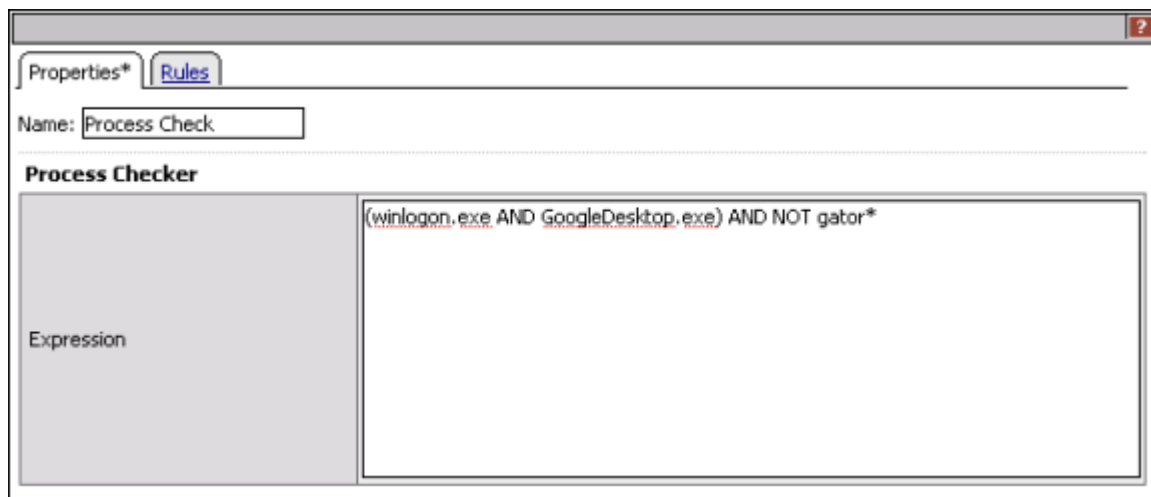
---

### ◆ Note

*This is not a complete example. For the example to work, you must assign an Allow ending to successful branches. You can assign a network access resource, portal access resources, app tunnels, remote desktops, and a webtop. For a web access management connection, you need not assign resources. This example is configured starting with an empty access policy.*

### To add the example action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a branch of the access policy, click the plus sign ( **+** ) to add an action.  
The Add Item popup screen opens.
4. On the Endpoint Security (Client-Side) tab, select **Windows Process** and click **Add Item** to add the action to the access policy.  
The Process Check action popup screen opens.
5. In the **Expression** box, type the process check expression as follows:  
  
**(winlogon.exe AND GoogleDesktop.exe) AND NOT gator\***  
  
The configured action appears as shown in Figure 7.4.
6. Click **Save** to complete the configuration.



*Figure 7.4 Process check action example*

## Setting up Windows registry check

You can set up the Windows registry action or verify the existence or absence of certain keys and values in the Windows system registry database. Both key values or Boolean expressions evaluate the existence or absence of registry entries.

### Expression syntax

Syntax for registry checker expressions is as follows:

**"key" comparison\_operator data**

**"key" ISPR**

**"key"."value" comparison\_operator data**

**"key"."value" ISPR**

- **"key"**  
Represents a path in the Windows registry.
- **"value"**  
Represents the name of the value.
- **comparison\_operator**  
Represents one of the comparison operators (< available <= **available** > **available** >= available =) or **ISPR**. **ISPR** is used to verify that a key or value is present.  
For equality use =. The operator == is not valid here.
- **data**  
Represents the content to compare against.

---

#### ◆ Note

*Quotation marks (") are required around **key** and **value** arguments. Quotation marks are used in **data** if the content contains spaces, commas, slashes, tabs, or other delimiters. If quotation marks exist as part of the registry path or value name, they should be doubled (use two sets of quotation marks). **data** is treated as a **version number** if it is entered in the format "**d.d[d].[d]**" or "**d,d[d],d]**" (where **d** is a number), and as a **date** if it is entered in the format "**mm/dd/yyyy**".*

### Specifying registry values

Following are examples of registry strings that you can use in the Registry action.

- **"HKEY\_CURRENT\_USER\Software\Microsoft\Windows\CurrentVersion\XP"**  
Checks for the presence of the specified path in the registry.
- **"HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Internet Explorer.Version">= "6.0.2900.2180"**  
Checks that the Internet Explorer version is greater than or equal to the value specified.

- `"HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\InternetExplorer.Version" >= "5.0.2800.0" AND "HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\InternetExplorer.Version" <= "6.0.2900.0"`  
Checks for the presence of Internet Explorer. With this registry check, the Internet Explorer version must be greater than or equal to **5.0.2800.0**, and less than or equal to **6.0.2900.0**.

## Specifying 32 and 64-bit registry keys on 64-bit Windows versions

On 64-bit Windows systems, you can check for registry keys in either the 64-bit registry or the 32-bit registry. To specify the registry to check, append a number to the registry root key name. The following key names are supported:

- HKEY\_CURRENT\_USER
- HKEY\_CURRENT\_USER32
- HKEY\_CURRENT\_USER64
- HKEY\_LOCAL\_MACHINE
- HKEY\_LOCAL\_MACHINE32
- HKEY\_LOCAL\_MACHINE64
- HKEY\_CLASSES\_ROOT
- HKEY\_CLASSES\_ROOT32HKEY\_CLASSES\_ROOT64
- HKEY\_USERS
- HKEY\_USERS32
- HKEY\_USERS64

HKEY values specified with a 32 allow you to check values in the 32-bit view of a 64-bit registry. This is the perspective used by 32-bit applications running on a 64-bit operating system.

HKEY values with a 64 appended allow you to check values in the 64-bit view of the registry. This is the perspective used by native 64-bit applications.

Keys without a bit value specified use the default Windows registry redirectors, as specified by Microsoft in the following article.

*Registry Redirector (Windows)*

([http://msdn.microsoft.com/en-us/library/aa384232\(VS.85\).aspx](http://msdn.microsoft.com/en-us/library/aa384232(VS.85).aspx))

When checking values on 32-bit Windows, the number of bits specified in the registry key name is ignored.

## Example: Using the Windows registry action

This example uses the registry checker to check for the presence of a Google Desktop resource DLL.

### To configure the example action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a branch of the access policy, click the plus sign ( **+** ) to add an action.  
The Add Item popup screen opens.
4. On the Endpoint Security (Client-Side) tab, select **Windows Registry** and click **Add Item** to add the action to the access policy.  
The Registry action popup screen opens.
5. In the **Expression** box, type:  
**"HKEY\_LOCAL\_MACHINE\Software\Google\GoogleDesktop.ResourceDLL"**  
The configured action appears as shown in Figure 7.5.
6. Click **Save** to complete the configuration.

The screenshot displays a configuration window for a 'Registry Checker' action. At the top, there are two tabs: 'Properties' and 'Branch Rules', with 'Branch Rules' being the active tab. Below the tabs, the 'Name' field is set to 'Registry Check'. Underneath, the title 'Registry Checker' is shown. The main area is divided into two sections: 'Expression' on the left and a text input field on the right. The text input field contains the registry path: "HKEY\_LOCAL\_MACHINE\Software\Google\GoogleDesktop.ResourceDLL".

*Figure 7.5 Registry check action example*

## Setting up Windows cache and session control

Use the cache and session control action to provide a higher level of security to systems that are logged on to your network. The cache and session control agent deletes browser cache and other session-related information, and can be configured to clean various settings from the user's system after a session is closed.

In an access policy, the cache and session control action is considered successful when the browser add-on starts successfully on the client computer. A failure indicates that the cache and session control action was unable to start.

---

### ◆ Note

*You can use the cache and session control action to clean cache and related session information from the Internet Explorer browser only. The action does not clear browser cache and session-related items from Firefox, Safari, or any other browser. However, other items you configure in the action are cleaned on all Windows systems.*

---

### ◆ Note

*Windows Cache and Session Control is not compatible with Windows Protected Workspace. You should not use a Windows Protected Workspace action in a session that includes the Windows Cache and Session Control action.*

## Setting up the cache and session control access policy item

Add a cache and session control action anywhere in the access policy, as long as it is used on a branch for Windows clients.

### To add a cache and session control action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a branch of the access policy, click the plus sign ( **+** ) to add an action.  
The Add Item popup screen opens.
4. On the Endpoint Security (Client-Side) tab, select **Windows Cache and Session Control** and click **Add Item** to add the action to the access policy.  
The Cache and Session Control action popup screen opens.



5. Configure the entry.

- For the option **Clean forms and passwords autocomplete data** option, select **Enabled** or **Disabled**.  
**Enabled** removes autocomplete data from web forms, and deletes saved passwords from the system after the user logs out.
- For the option **Empty Recycle Bin**, select **Enabled** or **Disabled**.  
**Enabled** ensures that the Recycle Bin is emptied on the system after the user logs out.
- For the option **Force session termination if the browser or Webtop is closed**, select **Enabled** or **Disabled**.  
**Enabled** forces the session to close when the user closes the web browser or the network access webtop.
- For the option **Remove dial-up entries used by Network Access client**, select **Enabled** or **Disabled**.  
**Enabled** removes the VPN connection from the user's Network Connections **Dial-up Networking** folder.
- From the **Terminate session on user inactivity** list, select a setting in minutes or hours to force the session to close if the user is inactive for the specified time.  
Select **Custom** to specify a custom setting, in seconds.  
Select **Disabled** to not terminate the session on user inactivity.  
User inactivity is the period of time during which the user has not input any data using the keyboard or mouse on the client system. This is not traffic inactivity over the VPN.
- From the **Lock workstation on user inactivity** list, select a setting in minutes or hours to force the user's workstation to lock if the user is inactive for the specified time.  
Select **Custom** to specify a custom setting, in seconds. Select **Disabled** to not lock the user's workstation because of user inactivity.  
User inactivity is the period of time during which the user has not input any data using the keyboard or mouse on the client system. This is not traffic inactivity over the VPN.

6. Click **Save** to complete the configuration.

## Example: Using cache and session control

In this example, the administrator adds a cache and session control that removes stored passwords and autocomplete data, forces the user to log out if the Webtop or browser is closed, locks the workstation after 5 minutes of inactivity, and closes any session that is inactive after 30 minutes. All other settings are left disabled.

### ◆ Note

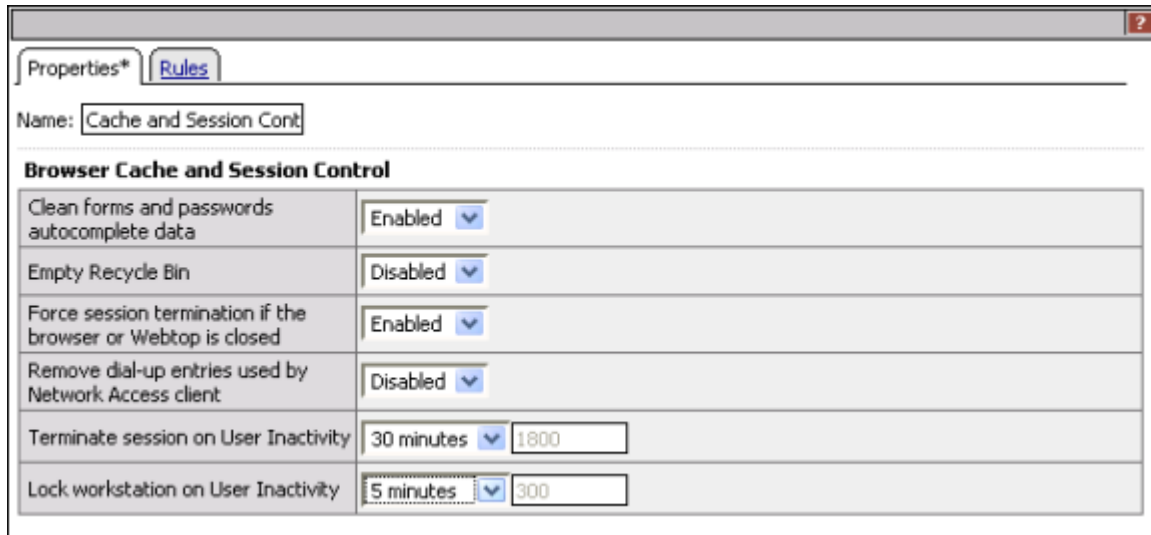
*This is not a complete example. For the example to work, you must assign an Allow ending to successful branches. You can assign a network access resource, portal access resources, app tunnels, remote desktops, and a webtop. For a web access management connection, you need not assign resources. This example is configured starting with an empty access policy.*

### To configure the example action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a branch of the access policy, click the plus sign ( **+** ) to add an action.  
The Add Item popup screen opens.
4. On the Endpoint Security (Client-Side) tab, select **Windows Cache and Session Control**, and click **Add Item** to add the action to the access policy.  
The Cache and Session Control action popup screen opens.
5. Configure the entry.
  - For the option **Clean forms and passwords autocomplete data**, select **Enabled**.
  - For the option **Force session termination if the browser or Webtop is closed**, select **Enabled**.
  - From the **Terminate session on user inactivity** list, select **30 minutes** to force the session to close after 30 minutes of inactivity.
  - From the **Lock workstation on user inactivity** list, select **5 minutes** to lock the user's workstation after 5 minutes of inactivity.

The completed policy appears as shown in Figure 7.6.

6. Click **Save** to complete the configuration.



Properties\* [Rules](#)

Name:

**Browser Cache and Session Control**

Clean forms and passwords autocomplete data	Enabled <input type="button" value="v"/>
Empty Recycle Bin	Disabled <input type="button" value="v"/>
Force session termination if the browser or Webtop is closed	Enabled <input type="button" value="v"/>
Remove dial-up entries used by Network Access client	Disabled <input type="button" value="v"/>
Terminate session on User Inactivity	30 minutes <input type="button" value="v"/> <input type="text" value="1800"/>
Lock workstation on User Inactivity	5 minutes <input type="button" value="v"/> <input type="text" value="300"/>

*Figure 7.6 Cache and session control action example*

## Setting up Windows protected workspace

Protected workspace configures a temporary Windows user workspace for the secure access session that prevents external access, and deletes any files created before leaving the protected area. The protected workspace allows you to restrict end users from printing and saving files on a client accessing the Access Policy Manager. Protected workspace reduces the risk of unintentional or accidental information leaks, but does not eliminate it. For example, **EXE**, **DLL**, and **IME** files are not encrypted. It restricts users to a temporary workspace on the remote system, which is newly created at the beginning of each new session. This workspace contains temporary **Desktop** and **My Documents** folders. In protected mode, the user cannot unintentionally or accidentally write files to locations outside the temporary folders. The protected workspace control deletes the temporary workspace and all of the folder contents at the end of the session.

---

### ◆ Note

*Windows Cache and Session Control is not compatible with Windows Protected Workspace. You should not use a Windows Protected Workspace action in a session that includes the Windows Cache and Session Control action.*

---


### ◆ Note

*You cannot assign a Windows group policy template after a session is in the protected workspace. To use Windows group policies with protected workspace, you must place the Windows group policy action before the protected workspace action in the access policy.*

## Setting up the protected workspace access policy item

Use the protected workspace action to assure that clients who connect to network access are placed in a protected workspace for the duration of the session.

### To add a protected workspace action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a branch of the access policy, click the plus sign (  ) to add an action.  
The Add Item popup screen opens.

4. On the Endpoint Security (Client-Side) tab, select **Windows Protected Workspace** and click **Add Item** to add the action to the access policy.  
The Protected Workspace action popup screen opens.
5. Configure the protected workspace.
  - Enable or disable the option to **Close Google Desktop Search** when the user starts the protected workspace session.  
Note that selecting **Enabled** in this option is more secure.
  - Enable or disable the option to **Allow user to temporarily switch from Protected Workspace** when the user is in the protected workspace session.
  - Enable or disable the option to **Allow user to use printers**.
  - Select the option for the setting **Allow write access to USB flash drives**. In addition to the **Disabled** option and the option to allow write access to **All USB flash drives**, this setting provides a third option, **Only IronKey Secure Flash Drives**, which allows a user to write only to specialized, highly secured flash drives created by IronKey, Inc.
  - Enable or disable the option to **Allow user to burn CDs**.
  - Enable or disable the option to **Allow user to choose storage location**. This specifies whether a user can choose the storage location for Protected Workspace files. **Enabled** allows users to select a storage location. **Disabled** stores files in the Document and Settings directory.
  - Select whether to **Enable persistent storage**. This specifies whether data is saved on the system after the protected workspace session is closed. **Enabled** allows users to save encrypted data from the protected workspace session on the local system after the session exits. The files are automatically decrypted and available in the next protected workspace session. **Disabled** prevents users from storing protected workspace data in persistent storage.
  - Select whether to **Password protect new storage**. Specifies whether protected workspace requires a password to access data in persistent storage. **Enabled** requires the user to set a password to access persistent storage data. **Disabled** uses the default encryption and decryption, which is based on the server group name and storage device volume serial number.
  - Specify a **Server group name**. This specifies a group name for the server. This name is arbitrary, but limits the persistent storage to that group name. For example, if a user connects to Protected Workspace on a server with group name **GroupA**, and persistent storage is enabled, the user data is available when reconnecting to a protected workspace session with the group name **GroupA**. However, if the user then connects to a server with persistent storage enabled and the server group name **GroupB**, persistent

data from the **GroupA** protected workspace session is not available in the new session, and a new persistent storage is defined.

6. If you want to allow protected workspace users to have write access to a specific server, click the **Add new entry** button and type the name of the server under **Allow write access to these servers**. To add more servers, repeat this step. To remove a server, click the **X** button next to the name of the server.
7. Click **Save** to complete the configuration.

## Example: Using protected workspace

In this example, the administrator adds protected workspace to an access policy branch. The security policy is very strict, so the only option allowed is for a user to write to an IronKey USB flash drive, and a server called Quarantine. Persistent storage is not enabled.

### ◆ Note

---

*This is not a complete example. For the example to work, you must assign an Allow ending to successful branches. You can assign a network access resource, portal access resources, app tunnels, remote desktops, and a webtop. For a web access management connection, you need not assign resources. This example is configured starting with an empty access policy.*

### To configure the example action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a branch of the access policy, click the plus sign ( **+** ) to add an action.  
The Add Item popup screen opens.
4. On the Endpoint Security (Client-Side) tab, select **Windows Protected Workspace** and click **Add Item** to add the action to the access policy.  
The Protected Workspace action popup screen opens.
5. Configure the action as follows:
  - From the **Close Google Desktop Search** list, select **Enabled**.
  - From the **Allow user to temporarily switch from Protected Workspace** list, select **Disabled**.
  - From the **Allow user to use printers** list, select **Disabled**.

- From the **Allow write access to USB flash drives** list, select **Only IronKey Secure Flash Drives**.
  - From the **Allow user to burn CDs** list, select **Disabled**.
  - From the **Allow user to choose storage location** list, select **Disabled**.
  - From the **Enable persistent storage** list, select **Disabled**.
  - From the **Password protect new storages** list, select **Enabled**.
  - Leave the **Server group name** list blank.
6. Click **Add new entry** to add a server to which a user can write. In the box that appears, type **Quarantine**. Note that new entries are added above previously configured entries, by default.
- The configured action appears as shown in Figure 7.7.
7. Click **Save** to save the access policy.

Properties\* [Branch Rules](#)

Name:

**Protected Workspace**

Close Google Desktop Search	Enabled ▾
Allow user to temporarily switch from Protected Workspace	Enabled ▾
Allow user to use printers	Enabled ▾
Allow write access to USB flash drives	Disabled ▾
Allow user to burn CDs	Disabled ▾
Allow user to choose storage location	Disabled ▾
Enable persistent storage	Disabled ▾
Password protect new storages	Enabled ▾
Server group name	<input type="text"/>

Insert Before: 1 ▾

Allow write access to these servers	
1	<input type="text" value="Quarantine"/>

**Figure 7.7** Protected workspace action example

## Assigning a Windows group policy template

The Windows group policy action allows you to assign a Windows group policy, which changes security settings for the Windows client environment for the duration of the network access session.

To use Windows group policy functionality, you must purchase a separate license for the feature.

### ◆ Note

*You cannot assign a Windows group policy template after a session is in the protected workspace. To use Windows group policies with protected workspace, you must place the Windows group policy action before the protected workspace action in the access policy.*

## Understanding Windows group policy templates

Windows group policy templates allow you to configure and assign group policies for Windows machines dynamically per user session in the access policy. Using Windows group policy templates, you can make configuration changes to client systems that exist for the duration of a session. The system applies Windows group policy changes after the Windows group policy check is successful, and before resources are assigned. After the user terminates the session, all Windows group policy changes are rolled back, and the client system reverts to its previous state.

You can use predefined Windows group policy templates with Access Policy Manager. To define your own Windows group policy templates, you must purchase a license for the GPAnywhere product from Full Armor.

## Using predefined Windows group policy templates

Table 7.1 lists the predefined Windows group policy templates included with Access Policy Manager, and their functional descriptions.

Template	Description
EC Domain XPSP2 Desktops Template	Microsoft Enterprise Client Policy for desktops and laptops. This is a moderate policy, balancing security and usability.
Firewall Settings Template	Access Policy Manager settings for enabling the user's firewall. This policy is used to ensure that the user's Microsoft firewall is configured and running.
GLBA Template	Based on the Gramm-Leach-Bliley GLBA standard. This policy is used for desktop and laptops to help prevent access to unauthorized information.
HIPAA Template	Based on the HIPAA (Health Insurance Portability and Accounting Act) standard. This policy is used for desktop and laptops to help prevent access to unauthorized information.

**Table 7.1** Predefined Windows group policy templates



Template	Description
Highly Managed Template	Microsoft Common Usage (high) for desktops and laptops. This policy is used in managed environments and provides high restrictions on user access to devices, configuration, and applications.
Lightly Managed Template	Microsoft Common Usage (light) for desktops and laptops. This policy is used in managed environments, and provides light restrictions on user access to devices, configuration, and applications.
PCI Template	Based on the PCI (Payment Card Industry) standard. This policy is used for desktop and laptops to help prevent access to unauthorized information.
SSLF Domain Template	Microsoft Specialized Security (Limited Functionality) for desktops and laptops. This is a more focused security policy, with greater restrictions on configuration access.
Terminal Services Taskstation Template	Terminal Services for client terminal services. This policy is used in environments where the primary use is terminal services.

**Table 7.1** *Predefined Windows group policy templates*

## Using the EC and SSLF templates

The Enterprise Client (EC) and Specialized Security—Limited Functionality (SSLF) templates are based on Microsoft security profiles for Enterprise Client and Specialized Security—Limited Functionality environments.

Microsoft uses the EC and SSLF environment classifications as the basis for making recommendations on how to configure a variety of server, workstation, and laptop settings. The EC Domain Template is applicable to most enterprise environments. It balances security with usability concerns. The Group Policy settings suggested for users in EC Domain-classified environments focus on addressing the basics at a moderate level, so it is not intrusive to the user.

Examples of settings that are applied as part of the EC Domain Template are:

- Disabling automatic saving of passwords in Internet Explorer
- Requiring that the user re-enter the password after a system suspend

The SSLF Domain Template is applicable to environments where concerns about security are paramount. In such an environment, some usability is sacrificed in order to further secure the systems. The Group Policy settings suggested for users in SSLF Domain-classified environments expand upon the settings recommended for the EC Domain.

Examples of settings that are applied as part of the SSLF Domain Template are:

- Disabling user access to the IE Security settings.
- Disabling user access to system tools such as the registry editor.

Additional information can be found in the Windows Server® 2003 security section at:

**<http://www.microsoft.com/technet/security/prodtech/windowsserver2003/w2003hg/s3sgch01.msp>**

## Using the Microsoft common scenario templates

Microsoft common scenarios classify client machines into categories such as mobile, multi-user, app-station, task-station, or kiosk. These scenarios are intended to provide common starting scenarios for group policy management.

### Understanding the managed templates

The highly- and lightly-managed templates are based on Microsoft Common Scenarios. To standardize the implementation of the scenarios, Microsoft defined the highly-managed and lightly-managed Group Policy settings as the base set of settings on top of which the scenarios would be implemented.

Both the lightly-managed and highly-managed policies are intended for use with devices that work in a centrally managed environment. As such, both templates restrict the options to which a user has access. The distinction between the two is a matter of degree.

In the case of the lightly-managed template, the users retain some ability to customize their desktop environment. Examples of settings that are applied as part of the lightly-managed template are:

- Enabling user access only to the Desktop Control Panel applet
- Prohibiting access to the Add/Remove Programs Windows Components page

In the case of the highly-managed template, the user is given very little leeway to customize the desktop environment. Examples of settings that are applied as part of the highly-managed template are:

- Prohibiting access to the Control Panel
- Denying access to Add/Remove Programs
- Prohibiting adding printers

For additional information, read Implementing Common Desktop Management Scenarios at:

**[http://technet.microsoft.com/en-us/library/cc758145\(WS.10\).aspx](http://technet.microsoft.com/en-us/library/cc758145(WS.10).aspx)**

### Understanding the terminal services task station template

The terminal services task station template is specific to terminal server users. It prevents users from reverting back to the default security policy but more importantly, it controls which file types (**.exe**, **.bat**, and **.msi**) can be used. While there are no restrictions on shortcuts (**.lnk**), restrictions are placed on the actual path of executables.

## Understanding the firewall settings template

The firewall settings template enables a user's firewall. This policy is used to ensure that the user's Microsoft firewall is configured and running. If the Microsoft Windows Firewall is not enabled, group policy starts it.

## Understanding the regulatory templates

The final three pre-configured templates help address certain regulatory requirements. They are all based on a basic security policy with their own nuances.

### Understanding the GLBA template

Gramm-Leach-Bliley Act (GLBA), also known as the Financial Services Modernization Act, enabled investment banks to merge with commercial banks and permitted insurance services to merge with securities companies. As part of this act, privacy policies are required to protect sensitive information from security threats. With GLBA, financial institutions must inform consumers, through a privacy notice, how the company collects, stores, shares, and safeguards the data. Compliance with the GLBA is mandatory for any financial services company.

Examples of settings that are applied as part of the GLBA template:

- Disabling CD-ROM and floppy drive access
- Digitally signing all communications, if available
- Prohibiting the user from modifying any certificate settings
- Prohibiting access to the Advanced Settings menu in Network Connections

### Understanding the HIPAA template

The Health Insurance Portability and Accountability Act (HIPAA) protects people with continued health insurance coverage if they lose or change jobs, and also establishes guidelines for the exchange of patient data, including electronic transmission. There are privacy rules for the use and disclosure of this patient information.

Examples of settings that are applied as part of the HIPAA template:

- Restricting CD-ROM access to locally logged-on users only.
- Prohibiting access to the Advanced Settings menu in Network Connections.
- Locking the workstation if the smartcard is removed.
- Clearing virtual memory.

### Understanding the PCI template

The Payment Card Industry Data Security Standard (PCI DSS) was designed by the major credit card companies as a guideline for any organizations that process credit card transactions. Like GLBA and HIPAA,

it establishes procedures for processing, storing, and transmitting sensitive data, and offers some protection against security vulnerabilities that may expose that information. Companies using PCI must also go through an outside audit to validate their compliance. There are 12 requirements within 6 major areas of concern: network security monitoring, network security testing, protecting cardholder data, vulnerability management, access control, and policy maintenance. You can find the specifics of PCI DSS at:

**[https://www.pcisecuritystandards.org/security\\_standards/pci\\_dss.shtml](https://www.pcisecuritystandards.org/security_standards/pci_dss.shtml)**

Examples of settings that are applied as part of the PCI template:

- Suspend session after 15 minutes of inactivity.
- Restrict anonymous access to Named Shares.
- Disable Advanced Settings in Internet Explorer.

## Working with Windows group policy templates

In addition to the preinstalled group policy templates explained above, you can add custom group policy templates, you can download templates installed on the Access Policy Manager, and you can view the configuration of installed templates.

### To add a Windows group policy template to the Access Policy Manager

1. On the Main tab of the navigation pane, expand **Access Policy**.
2. Hover your mouse pointer over **Access Profiles**, and click the **Windows Group Policy** link that appears.  
The Windows Group Policy List screen opens.
3. Click **Create**.  
The New Windows Group Policy screen opens.
4. In the **Name** box, type a name for the group policy.
5. In the **Description** box, type an optional description of the group policy.  
This description appears on the Windows Group Policy List screen, in the Description column.
6. In the **Configuration File** box, click **Browse** to locate the file.  
Configuration files are created by the FullArmor GPAnywhere product, and are Windows executable files with an **EXE** extension.
7. Click **Finished** when the configuration is complete.

### To download a Windows group policy template

1. On the Main tab of the navigation pane, expand **Access Policy**.
2. Hover your mouse pointer over **Access Profiles**, and click the **Windows Group Policy** link that appears.  
The Windows Group Policy List screen opens.

3. Click the group policy template that you want to download.  
The template Properties screen opens.
4. Next to **Configuration File**, click the **Download** link.  
The web browser pops up a **Save file** dialog.
5. Click the **Save** button to save the file.


### To view a Windows group policy template

1. On the Main tab of the navigation pane, expand **Access Policy**.
2. Hover your mouse pointer over **Access Profiles**, and click the **Windows Group Policy** link that appears.  
The Windows Group Policy List screen opens.
3. Click the group policy template that you want to download.  
The template Properties screen opens.
4. Next to **Configuration Details**, click the **View** link.  
The web browser pops up a save file dialog.
5. Save the file.

## Setting up the Windows group policy access policy item

Use the Windows group policy action to assure that clients who connect to network access have their computers configured to conform to the security policy required for the duration of the session.

### To add a Windows group policy action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a branch of the access policy, click the plus sign (  ) to add an action.  
The Add Item popup screen opens.
4. On the Endpoint Security (Client-Side) tab, select **Windows Group Policy** and click **Add Item** to add the action to the access policy.  
The Windows group policy action popup screen opens.
5. From the Windows group policy list, select the group policy to apply to client computers.  
You can add your own group policy templates that you create with the FullArmor GPAnywhere add-on. For more information on group policy templates, see *Understanding Windows group policy templates*, on page 7-32.

6. Click **Save** to complete the configuration.


## Example: Using Windows group policy templates

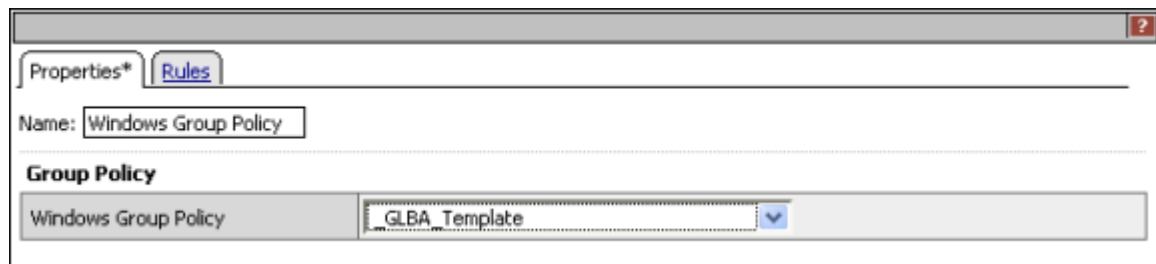
In this example, the administrator adds the predefined Gramm-Leach-Bliley Act (GLBA) Windows group policy template to clients that connect through this branch on the access policy. The Gramm-Leach-Bliley Act requires financial institutions to inform consumers, through a privacy notice, how the company collects, stores, shares, and safeguards the data. GLBA is mandatory for any financial services company.

### ◆ Note

*This is not a complete example. For the example to work, you must assign an Allow ending to successful branches. You can assign a network access resource, portal access resources, app tunnels, remote desktops, and a webtop. For a web access management connection, you need not assign resources. This example is configured starting with an empty access policy.*

### To configure the example action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a branch of the access policy, click the plus sign (  ) to add an action.  
The Add Item popup screen opens.
4. On the Endpoint Security (Client-Side) tab, select **Windows Group Policy** and click **Add Item** to add the action to the access policy.  
The Windows group policy action popup screen opens.
5. From the Windows Group Policy list, select **\_GLBA\_Template**.  
The configured action appears as shown in Figure 7.8.
6. Click **Save** to save the access policy.



*Figure 7.8 Windows group policy action example*

## Checking software on an endpoint

You can check a client system for the presence and condition of software using these access policy actions that perform software checks:

- Antivirus
- Anti-Spyware
- Firewall
- Hard Disk Encryption
- Patch Management
- Peer-to-peer
- Windows Health Agent

When you configure properties for these endpoint security software checks, you can specify what you want to check from the presence of any software to particular vendors and versions.

Not all software checks are available on all supported platforms. The properties screen for a software check provides a **Platform** setting when multiple platforms are supported.

## About supported vendor and product ID lists in software checks

When you configure a software check, you can generally select from lists of supported vendors and products. The contents of these lists depend upon the version of the EPSEC package that you have installed on your BIG-IP system. To view the currently supported software, click the **OPSWAT application integration support charts** link on the BIG-IP system start page.

## About recurring endpoint checks

Software checks include this setting: **Continuously check the result and end the session if it changes**. It is disabled by default. Set it to **Enabled** to continuously check the software on the client endpoint and terminate the session if the result changes.





# 8

---

## Configuring Endpoint Security (Server-Side)

---

- Introducing endpoint security (server-side) checks
- Configuring client OS check
- Configuring cliweb site at ent type check
- Checking for client-side check capability
- Checking a landing URI with the landing URI check
- Identifying Microsoft Exchange clients with the client for MS Exchange check
- Using IP Geolocation in an access policy



## Introducing endpoint security (server-side) checks

In addition to client-side checks, the BIG-IP® Access Policy Manager® provides server-side checks. When the access policy is processed, server-side checks allow the server to check clients and make policy decisions based on information that a client presents to the server. For example, the client type check presents a query to find out what type of client is connecting, and routes the client to the different policy branches based on the results of the query.

For a complete list and brief descriptions of endpoint security (server-side) checks, see *Understanding available actions*, on page 4-3. The basic process for adding an action to an access policy is the same for each action. Properties and branch rules differ from action to action.

## Preparing for clients that cannot use client checks

The administrator can configure an access policy to provide access for non-Windows clients, or clients that do not have the ability to install browser add-ons. To do this, the administrator adds a client-side check capability action at the start of the access policy, and then adds the client-side checks only on the **Full** access policy branch.

## Checking the landing URI of a client

The landing URI action checks the landing URI the client entered to reach the access policy. The landing URI is the actual landing address after the domain name; for example, for a Microsoft Outlook Web Access connection at <http://www.siterequest.com/owa>, the landing URI is **/owa**. The landing URI action provides a separate rule branch for each configured URI, and a fallback branch is provided for URIs that do not conform. For details, refer to *Checking a landing URI with the landing URI check*, on page 8-11.

## Configuring client OS check


The client OS check allows you to verify which operating system the client is using. The default client OS check includes eight rule branches. Seven of these rule branches correspond to the operating systems specified in the name of the rule. If, while running the access policy, Access Policy Manager® detects the operating system on the client as one of the specified operating systems, the access policy uses that rule branch. The access policy uses the fallback rule branch when it detects any other operating system. These are the operating system rule branches:

- Windows® (includes Windows version 7 and 8, Windows Server® 2008, Windows Vista®, Windows Server 2003, Windows XP, Windows 2000, and Windows NT)
- Windows RT  
**Note:** The Windows RT branch is available only when you have the appropriate Access Policy Manager 11.4.x hotfix installed. To determine hotfix requirements, refer to the BIG-IP APM Client Compatibility Matrix for APM 11.4.0 or APM 11.4.1 on the AskF5™ web site at <http://support.f5.com>.
- Linux
- Mac OS
- iOS
- Android

## Setting up the client OS check

We recommend that you use the client OS check at the beginning of an access policy, so you can build access policies using the separate operating system branches for functionality specific to those operating systems.

### To add a client OS action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a branch of the access policy, click the plus sign (  ) to add an action.  
The Add Item popup screen opens.
4. On the Endpoint Security (Server-Side) tab, select **Client OS** and click **Add Item** to add the action to the access policy.  
The Client OS action popup screen opens.

5. Click **Save** to complete the configuration.
6. To activate the access policy, click the **Apply Access Policy** link at the top of the visual policy editor screen.

## Example: Using client OS check

In this example, you add the client OS check to an access policy, and only the Windows, MacOS, iOS, and Android branches are assigned allowed endings.

### ◆ Note

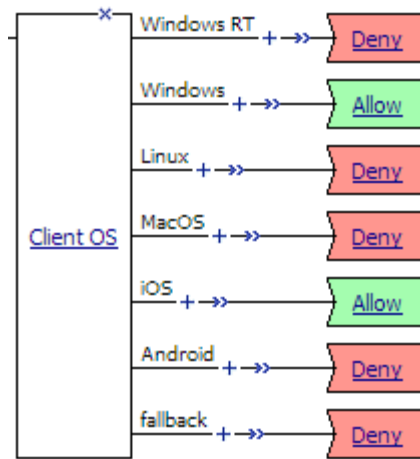
*This is not a complete example. For the example to work, you must assign an Allow ending to successful branches. You can assign a network access, portal access, app tunnel, or remote desktop resource using one of the resource assign actions, along with an associated network access, portal access, or full webtop. For a web access management connection, you need not assign resources. This example is configured starting with an empty access policy.*

### To add the example client OS check action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a branch of the access policy, click the plus sign ( **+** ) to add an action.  
The Add Item popup screen opens.
4. On the Endpoint Security (Server-Side) tab, select **Client OS** and click **Add Item** to add the action to the access policy.  
The Client OS action popup screen opens.
5. Click **Save**.
6. On the Windows, MacOS, iOS, and Android branches following the client OS action, configure allowed endings. Configure logon denied endings for all other branches.  
To configure endings, see *Configuring access policy endings*, on page 5-10.

The completed policy appears as shown in Figure 8.1.

7. To activate the access policy, click the **Apply Access Policy** link at the top of the visual policy editor screen.



*Figure 8.1 Client OS access policy example*

◆ **Note**

*The Windows RT branch shown in Figure 8.1 is available only when you have the appropriate Access Policy Manager® 11.4.x hotfix installed. To determine hotfix requirements, refer to the BIG-IP APM Client Compatibility Matrix for APM 11.4.0 or APM 11.4.1 on the AskF5™ web site at <http://support.f5.com>.*

## Configuring cliweb site at ent type check

You can use the client type check to determine whether the client is using a full browser, the standalone client, or another client to access the Access Policy Manager®. The default Client Type check includes six branches:


- An **Edge Portal®** branch, which indicates that the user is connecting with the Edge Portal mobile app.
- An **Edge Client®** branch, which indicates that the user is connecting with the BIG-IP® Edge Client or BIG-IP Edge Client app, supported on multiple devices and operating systems.
- A **Citrix Receiver** branch, which indicates that the user is connecting using a later Citrix receiver.
- A **Citrix Receiver (legacy)** branch, which indicates that the user is connecting using an earlier Citrix receiver.
- A **VMware View** branch, which indicates that the user is connecting using a VMware View client.
- A **Full or Mobile Browser** branch, which indicates that the user is connecting with a Windows web browser or a mobile browser.
- A **Windows® Built-in Client** branch, which indicates that the user is connecting from a Windows client using the Inbox F5® VPN Client.  
**Note:** This branch is available only when the appropriate Access Policy Manager version 11.4.x hotfix is installed. To determine hotfix requirements, refer to the BIG-IP APM® Client Compatibility Matrix for APM 11.4.0 or APM 11.4.1 on the AskF5™ web site at <http://support.f5.com>.
- A **fallback** branch, which indicates that the user is connecting with another method.

## Setting up the client type access policy item

We recommend that you use the client type check as one of the first checks in your access policy. You can then configure the **Edge Client®** branch with all of the checks that you require for fully capable clients, while also providing access policy branches for other clients. You can also provide different resources or simpler checks for mobile clients using the Edge Portal® app, and make other choices based on the client type response.

### To add a client type action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.

3. On a branch of the access policy, click the plus sign (  ) to add an action.  
The Add Item popup screen opens.
4. On the Endpoint Security (Server-Side) tab, select **Client Type** and click **Add Item** to add the action to the access policy.  
The Client Type action popup screen opens.
5. Click **Save** to complete the configuration.
6. To activate the access policy, click the **Apply Access Policy** link at the top of the visual policy editor screen.



## Example: Using client type check

In this example, you add a client type check, add a Windows cache and session control endpoint security check to the full browser branch, and change endings to allow for all non-fallback branches.

### ◆ Note

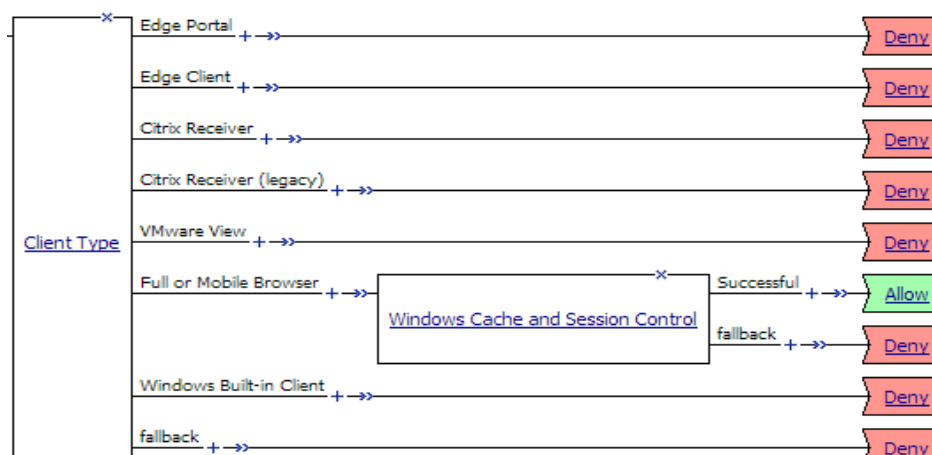
*This is not a complete example. For the example to work, you must assign an Allow ending to successful branches. You can assign a network access, portal access, app tunnel, or remote desktop resource using one of the resource assign actions, along with an associated network access, portal access, or full webtop. For a web access management connection, you need not assign resources. This example is configured starting with an empty access policy.*

### To add the example client type check action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a branch of the access policy, click the plus sign (  ) to add an action.  
The Add Item popup screen opens.
4. On the Endpoint Security (Server-Side) tab, select **Client Type** and click **Add Item** to add the action to the access policy.  
The Client Type action popup screen opens.
5. Click **Save**.
6. On the Full Browser branch following the Client Type action, click the plus sign (  ).  
The Add Item popup screen opens.



7. On the Endpoint Security (Server-Side) tab, select **Windows Cache and Session Control** and click **Add Item**.  
The cache and session control action popup screen opens.
8. Click **Save**.
9. On all branches except for the fallback branches, configure Allow endings.
10. Configure logon denied endings for all other branches.  
To configure endings, see *Configuring access policy endings*, on page 5-10.  
The completed policy appears as shown in Figure 8.2.
11. To activate the access policy, click the **Apply Access Policy** link at the top of the visual policy editor screen.



**Figure 8.2** Client Type access policy example

#### ◆ Note

The Windows Built-in Client branch shown in Figure 8.2 is available only when you have the appropriate Access Policy Manager® version 11.4.x hotfix installed. To determine hotfix requirements, refer to the BIG-IP APM Client Compatibility Matrix for APM 11.4.0 or APM 11.4.1 on the AskF5™ web site at <http://support.f5.com>.

## Checking for client-side check capability


You can use the client-side check capability action to determine whether the client has the ability to run client-side checks. The default endpoint check capability action includes two branches:

- A **Full** branch, which indicates that the user is connecting with a client that has full client-side check support.
- A **Fallback** branch, which indicates that the user is connecting with a client that does not fully support client-side checks.

## Setting up the client-side check capability access policy item

We recommend that you use the client-side check capability action as one of the first checks in your access policy. You can then configure the **Full** branch with all of the endpoint security checks that you require for your endpoint-security capable clients, while also providing access policy branches for other clients.

### To add a client-side check capability action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a branch of the access policy, click the plus sign (  ) to add an action.  
The Add Item popup screen opens.
4. On the Endpoint Security (Server-Side) tab, select **Client-Side Capability** and click **Add Item** to add the action to the access policy.  
The Client-Side Capability action popup screen opens.
5. Click **Save** to complete the configuration.
6. To activate the access policy, click the **Apply Access Policy** link at the top of the visual policy editor screen.

## Example: Using client-side check capability action

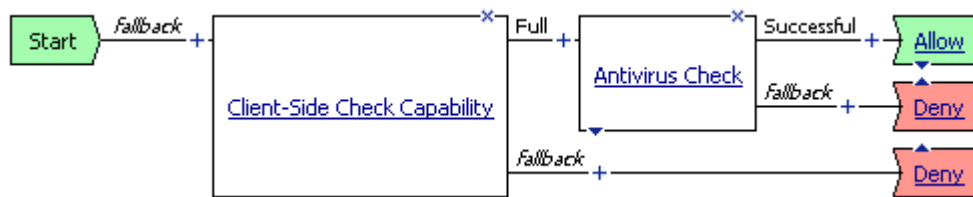
In this example, you add a client-side check capability action, then add an antivirus client-side check to the **Full** branch.

### ◆ Note

*This is not a complete example. For the example to work, you must assign an Allow ending to successful branches. You can assign a network access, portal access, app tunnel, or remote desktop resource using one of the resource assign actions, along with an associated network access, portal access, or full webtop. For web access management connection, you need not assign resources. This example is configured starting with an empty access policy.*

### To add the example client-side check capability action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a branch of the access policy, click the plus sign ( **+** ) to add an action.  
The Add Item popup screen opens.
4. On the Endpoint Security (Server-Side) tab, select **Client-Side Capability** and click **Add Item** to add the action to the access policy.  
The Client-Side Capability action popup screen opens.
5. Click **Save**.
6. On the Full branch following the Client-Side Capability action, click the plus sign ( **+** ).  
The Add Item popup screen opens.
7. On the Endpoint Security (Client-Side) tab, select **Antivirus** and click **Add Item**.  
The antivirus action popup screen opens.
8. Click **Save**.
9. On the **Successful** branch following the antivirus action, configure an Allow ending.
10. Configure logon denied endings for all other branches.  
To configure endings, see *Configuring access policy endings*, on page 5-10.  
The completed policy appears as shown in Figure 8.3.
11. To activate the access policy, click the **Apply Access Policy** link at the top of the visual policy editor screen.



**Figure 8.3** Client-side check capability access policy example

## Checking a landing URI with the landing URI check


You can use the Landing URI check to check the landing URI with which the user has accessed the access policy. The default Landing URI check includes two branches:

- A **Landing URI** branch, which indicates the landing URI for which the policy should check, and evaluates as true if the specified landing URI is reached.
- A **Fallback** branch, which indicates that the user is connecting with a different landing URI.

## Setting up the landing URI access policy item

We recommend that you use the landing URI check to determine the landing URI that the user typed to connect to the Access Policy Manager®.

### To add a landing URI action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a branch of the access policy, click the plus sign (  ) to add an action.  
The Add Item popup screen opens.
4. On the Endpoint Security (Server-Side) tab, select **Landing URI** and click **Add Item** to add the action to the access policy.  
The Landing URI action popup screen opens.
5. Click **Save** to complete the configuration.
6. To activate the access policy, click the **Apply Access Policy** link at the top of the visual policy editor screen.

## Example: Using landing URI check

In this example, your Microsoft Outlook Web Access address is **http://www.siterequest.com/owa**. You add a landing URI check that checks for the landing URI **/owa**, the typical landing URI for an Outlook Web Access connection. If the access policy finds this URI, you can then add a resource assign action on the successful policy branch. In this

example, you add a resource assign action after the landing URI check for the URI **/owa**. For a complete working scenario, assign a portal access resource for Outlook Web Access with this resource assign action.

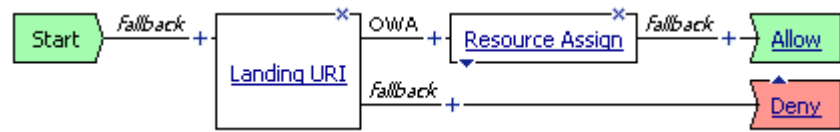
◆ **Note**

---

*This example does not detail how to create and assign portal access resources. For detailed instructions, see the **BIG-IP® Access Policy Manager® Portal Access Guide**, and *Assigning resources*, on page 6-8.*

### **To add the example Landing URI check action**

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a branch of the access policy, click the plus sign ( **+** ) to add an action.  
The Add Item popup screen opens.
4. On the Endpoint Security (Server-Side) tab, select **Landing URI** and click **Add Item** to add the action to the access policy.  
The Landing URI action popup screen opens.
5. In the **Name** box, type **owa**.
6. Click the Rules tab.  
The Rules for the action popup screen are displayed. The predefined rule for this action is **Expression: Landing URI is /uri1**.
7. Next to **Expression: Landing URI is /uri1**, click the **change** link.  
The expression builder popup screen opens.
8. In the **Landing URI is** box, type **/owa**.  
On the **OWA** branch, add a resource assign action and configure it for Outlook Web Access, if you have an Outlook Web Access server and resources.
  - To configure the web application, see the **BIG-IP® Access Policy Manager® Portal Access Guide**.
  - To assign the resource, see *Assigning resources*, on page 6-8.  
The completed policy appears as shown in Figure 8.4.
9. Click **Save**.
10. To activate the access policy, click the **Apply Access Policy** link at the top of the visual policy editor screen.



**Figure 8.4** Landing URI access policy example

## Identifying Microsoft Exchange clients with the client for MS Exchange check

You can use the client for MS Exchange check to determine if a client is using Microsoft Exchange or ActiveSync protocols. The default client for MS Exchange check includes two branches:

- A **Client for MS Exchange** branch.
- A **Fallback** branch, which indicates that the user is not using MS Exchange or ActiveSync.

The client for MS Exchange check requires that you also add an Exchange profile to the access profile.

## Understanding Microsoft Exchange connections

A client for MS Exchange is not a typical web browser, and Access Policy Manager® has the following restrictions on MS Exchange access policy branches.

- The MS Exchange client branch cannot provide responses that require additional user input, except for the logon page.
- Authentication retries are not attempted.
- You must assign a logon page action to the access policy. The logon page action will automatically work in clientless mode.

MS Exchange devices support only the following actions, and you should not use other actions on a client for MS Exchange branch:

- Active Directory Authentication
- Active Directory Query
- Client Certificate Inspection
- HTTP Authentication
- LDAP Authentication
- LDAP Query
- NTLM Authentication
- RADIUS Authentication
- RADIUS Accounting
- RSA SecurID Authentication
- Client-Side Capability
- Client OS
- Landing URI
- IP Geolocation Match

The following actions are not supported on MS Exchange clients:



- On-Demand Certificate Authentication
- any client side check

## Setting up the MS Exchange check policy item

We recommend that you use the MS Exchange check to determine when a user is connecting with an Exchange or ActiveSync client.

### To add a client for MS Exchange check

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a branch of the access policy, click the plus sign ( **+** ) to add an action.  
The Add Item popup screen opens.
4. On the Endpoint Security (Server-Side) tab, select **Client for MS Exchange** and click **Add Item** to add the action to the access policy.  
The Client for MS Exchange action popup screen opens.
5. Click **Save** to complete the configuration.
6. To activate the access policy, click the **Apply Access Policy** link at the top of the visual policy editor screen.

## Example: Using client for MS Exchange check

In this example, for a complete working scenario, assign a portal access resource for Outlook Web Access with this resource assign action. Note that you must also add an Exchange profile to the access profile.

---

### ◆ Note

*This example does not detail how to create and assign portal access resources. For detailed instructions, see the **BIG-IP® Access Policy Manager® Portal Access Guide**, and *Assigning resources*, on page 6-8.*

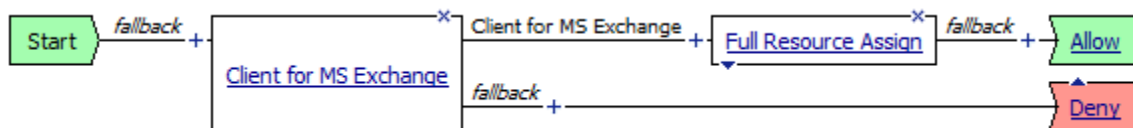
### To add the Client for MS Exchange check

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.

2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a branch of the access policy, click the plus sign ( **+** ) to add an action.  
The Add Item popup screen opens.
4. Select **Client for MS Exchange** and click **Add Item** to add the action to the access policy.  
The Client for MS Exchange action popup screen opens.
5. Click **Save**.
6. On the Client for MS Exchange branch, click the plus sign ( **+** ) to add an action.
7. Select **Advanced Resource Assign** and click **Add**.  
The Advanced Resource Assign action popup screen opens.
8. Click **Add new entry**
9. Under **Expression: empty** click **Add/Delete**.
10. Select Portal Access Resources from the menu tabs and select an Outlook Web Access resource.  
*You must predefine an Outlook Web Access portal access resource to select. You can create one manually or use the Portal Access wizard.*
11. Select Webtop from the menu tabs and select a portal access or full webtop.  
*You must predefine a portal access resource access or full webtop to select.*
12. Click **Save**.

The completed policy appears as shown in Figure 8.5.

To activate the access policy, click the **Apply Access Policy** link at the top of the visual policy editor screen.



**Figure 8.5** Client for MS Exchange access policy example

## Using IP Geolocation in an access policy

You can use the IP Geolocation match access policy item to make policy decisions based on geolocation by client IP address.

The IP geolocation match access policy item checks the client IP address against the geolocation database to determine the client's physical location. The available conditions are:


- **IP Geolocation Continent code is:** Specifies that the user's IP address must match the specified continent code.
- **IP Geolocation Country code is:** Specifies that the user's IP address must match the specified country code.
- **IP Geolocation Country name is:** Specifies that the user's IP address must match the specified country name.
- **IP Geolocation State/Region is:** Specifies that the user's IP address must match the specified region or state.

If the geolocation information determined from the IP address does not match the specified conditions, the access policy sends the user to the fallback branch.

## Setting up the IP geolocation match access policy item

We recommend that you use the IP geolocation match check to determine a user's physical location and make appropriate policy decisions based on that information.

### To add an IP geolocation match check

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a branch of the access policy, click the plus sign (  ) to add an action.  
The Add Item popup screen opens.
4. On the Endpoint Security (Server-Side) tab, select **IP Geolocation Match** and click **Add Item** to add the action to the access policy.  
The IP Geolocation Match action popup screen opens.  
  
The default setting for the IP geolocation match access policy item is to check that the country code for the IP address is US.
5. To make changes to the IP Geolocation Match settings, click the **Branch Rules** tab.

6. After you make changes, click **Save** to complete the configuration.
7. To activate the access policy, click the **Apply Access Policy** link at the top of the visual policy editor screen.

## Example: Using IP geolocation

In this example, you check that the IP address geolocation is for the United States and for the state of Texas, and assign resources to the successful branch. The fallback branch gets assigned a deny ending. For a complete working scenario, assign resources with the resource assign action.

---

### ◆ Note

*This example does not detail how to create and assign resources. For detailed instructions, see [Assigning resources](#), on page 6-8.*

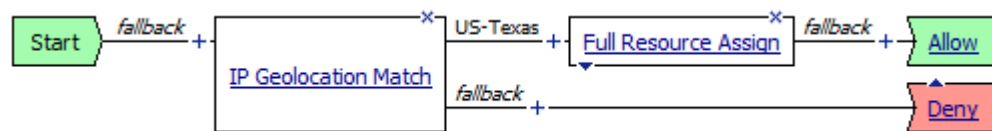
### To add the example IP geolocation match action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a branch of the access policy, click the plus sign ( **+** ) to add an action.  
The Add Item popup screen opens.
4. On the Endpoint Security (Server-Side) tab, select **IP Geolocation Match** and click **Add Item** to add the action to the access policy.  
The IP Geolocation Match action popup screen opens.
5. Click the **Branch Rules** tab.  
The **Branch Rules** screen opens.
6. In the **Name** field type **US-Texas**.
7. Next to **Expression: Country code: US**, click the **change** link.
8. Next to **AND**, click the **Add Expression** button.  
The **Add Expression** screen opens.
9. From the **Agent Sel** list, select **IP Geolocation Match**.
10. From the **Condition** list, select **IP Geolocation State/Region is**.
11. In the **State/Region** field, type **Texas**.
12. Click **Add Expression**.
13. Click **Finished**.
14. Click **Save**.

15. On the US-Texas branch, add a resource assign action and configure it for the policy assignment.

The completed policy appears as shown in Figure 8.6.

To activate the access policy, click the **Apply Access Policy** link at the top of the visual policy editor screen.



**Figure 8.6** IP geolocation match example action





# 9

---

## Using Certificate Authentication in APM

---

- Controlling SSL traffic
- Understanding SSL profiles
- Introducing SSL server certificates
- Understanding APM certificate authentication agents
- Configuring client SSL profiles
- Using certificates to authenticate users
- Understanding certificate revocation status
- Using CRLDP





## Controlling SSL traffic

One of the primary ways that you can control SSL network traffic is by configuring a client or server SSL profile. This chapter provides information on any features specific to Access Policy Manager® that you are required to configure to manage the client side, and ensure that your SSL certificate is set up properly for validation and authentication.

For more detailed information about managing SSL traffic, refer to available **BIG-IP® Local Traffic Manager™: Concepts** available on the AskF5™ web site at <http://support.f5.com>.

## Understanding SSL profiles

A **profile** is a group of settings with values that determine the way that the Access Policy Manager system handles application-specific network traffic. One type of traffic that a profile can manage is SSL traffic. The most basic functions of an SSL profile are to offload the certificate validation and verification tasks, as well as data encryption and decryption, from your targeted web servers. The two types of SSL profiles are:

- **Client Profiles**

Client Profiles allow the BIG-IP® system to handle authentication and encryption tasks for any SSL connection coming into a Access Policy Manager system from a client system. You implement this type of profile by using the default **clientssl** profile, or by creating a custom profile based on the default **clientssl** profile. For more information on how to set up an SSL profile for a client, refer to *Configuring client SSL profiles*, on page 9-8.

- **Server Profiles.**

Server Profiles allow the BIG-IP® system to handle encryption tasks for any SSL connection being sent from a Access Policy Manager to a target server. An SSL server profile is able to act as client by presenting certificate credentials to a server when authentication of the Access Policy Manager system is required. You implement this type of profile by using the default **serverssl** profile, or by creating a custom profile based on the default **serverssl** profile.

For more information about configuring server SSL profiles, refer to the **BIG-IP® Local Traffic Manager™: Concepts** guide available on the AskF5™ web site at <http://support.f5.com>.

## Introducing SSL server certificates

The SSL (**Secure Sockets Layer**) protocol uses the certificate to establish a secure connection. A valid SSL server certificate, also known as a security certificate, is necessary for establishing secure HTTPS connections. An **SSL server certificate** identifies your server to any connecting client browser.

The certificate contains information identifying the server, and the organization it was issued to, as well as an expiration date. Most browsers that support SSL connections have internal lists of Certificate Authorities (CAs), and automatically accept certificates issued by these organizations. If there is an error, some browsers display security warnings; other browsers, notably those found on wireless devices such as PDAs or smart phones, might refuse a connection.

When a client presents a certificate to the BIG-IP® system, the system uses a trusted CA file to determine the Certificate Authorities that it can trust. By using this file, the BIG-IP system attempts to verify a client certificate.

When you create an SSL client profile, as described in *Configuring client SSL profiles*, on page 9-8, the BIG-IP system automatically creates a default client trusted CA file.

For more detailed information about server and client side certificates, refer to **BIG-IP® Local Traffic Manager™: Concepts** guide available on the AskF5™ web site at <http://support.f5.com>.

## Understanding APM certificate authentication agents

The Access Policy Manager provides these types of agents for certificate authentication:

- ◆ Agents that validate client SSL certificates:
  - Client certificate inspection agent - Validates the result of the initial SSL handshake.
  - On-Demand Certificate authentication agent - Renegotiates the SSL handshake and validates the result.
- ◆ Agents that check certificate revocation status - For more information, see *Understanding certificate revocation status*, on page 9-11.

### Client certificate inspection agent

This Client certificate inspection agent checks the result of the SSL handshake request that occurs at the start of the session. It does not, however, negotiate an SSL session.

F5 Networks recommends that you use the Client certificate inspection agent in cases where certificate authentication is required as part of the initial SSL handshake, and only if it is necessary to validate certificate authentication as part of running the access policy.

The following example describes a Client certificate inspection agent being used as part of an access policy.

- The Client Certificate setting, **request**, in the **clientssl** profile, prompts the system to send a certificate authentication request to the user.
- After the user provides a valid certificate, the access policy is started by the system, and the system provides the logon page (the first item in the access policy). Note that the opening of the logon page agent is not affected by the result of the certificate authentication process.
- The RADIUS authorization agent (the second item in the access policy) authenticates the user.
- The Client certificate inspection starts upon successful authentication.
- The Client certificate inspection agent checks the result of the certificate authentication that was performed at the beginning of the session before the logon page agent.
- The default rule that comes with the client certification inspection agent checks the value of the session variable **session.ssl.cert.valid** to determine the success or failure of the authentication process. Upon successful authentication, the access policy assigns the resource **R1** to the user and reaches the allow ending. Otherwise, the access policy assigns the resource **R2** to the user.

To use this agent, set the **Client Certificate** setting in the **clientssl** profile to **request**; with this setting, a certificate request is sent to the client. In this case, the SSL profile always grants access, regardless of the status or

absence of the certificate. Granting access is not dependent on whether a certificate is present, nor does connection terminate if a certificate is not received.

◆ **Note**

---

*For **Client Certificate**, **request** is the recommended setting. Alternatively, you can use the **require** setting; however, if you do, the user must provide a valid client certificate. Otherwise, the connection is not allowed*

## On-Demand certificate authentication agent

The On-Demand certificate authentication agent performs an SSL re-handshake and validates the received certificate. To use this agent, select **ignore** for the **Client Certificate** setting in the **clientssl** profile on the New Client SSL Profile screen. The system disregards the certificate request and does not use it in the initial SSL handshake.

We recommend that you use this agent in cases where both certificate authentication and validation need to be performed in the middle of an access policy process.

The following example describes an On-Demand certificate authentication agent being used as part of the access policy.

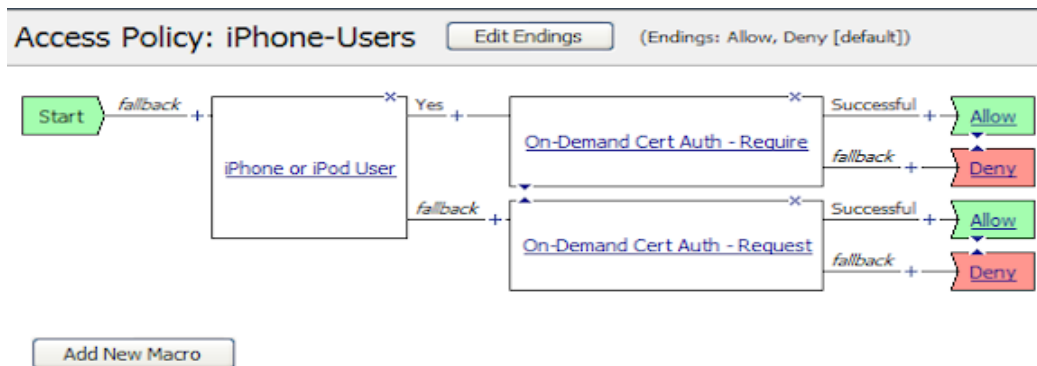
- When the user connects to the system, with the **Client Certificate** setting in the **clientssl** profile set to **ignore**, the system does not prompt a request to the user for a certificate, but instead the access policy process starts by providing the Logon page to the user.
- After the user enters his credentials, the RADIUS authentication agent starts.
- Upon successful authentication, the access policy runs the decision box action called **Client Cert Installed or Not**, which prompts the user to indicate whether he has a certificate installed.
- If the user selects **Yes**, then the On-Demand certificate authentication agent runs.
- The On-Demand certificate agent re-negotiates the SSL connection by sending a certificate request to the user, which prompts a certificate window to open.
- After the user provides a valid certificate, the On-Demand certificate authentication agent checks the result of the certificate authentication. The default rule that comes with the On-Demand certificate authentication agent checks the value of the session variable **session.ssl.cert.valid** to determine whether authentication was a success.
- If the access policy rule in the On-Demand certificate authentication agent detects that the validation was a success, then the access policy assigns the resource **R1** to the user, and takes the user to the allow ending. Otherwise, the user is denied access.

## Configuring the On-Demand Cert Auth agent Auth Mode property

When you add the On-Demand certificate authentication agent to an access policy, you can configure the **Auth Mode** property for it. There are two possible values:

- **request**: The system requests a valid certificate from the client, but the connection does not terminate if the client does not provide a valid certificate. Instead, this action takes the fallback route in the access policy. This is the default option.
- **require**: The system requires that a client provides a valid certificate. If the client does not provide a valid certificate, the connection terminates and the client browser stops responding.

Figure 9.1 shows an example of an access policy that displays the **On-Demand Cert Auth** agent with the two authentication modes. The **iPhone or iPod User** check is created using the Client Type check, and determines whether the client is using an iPhone or iPod, or the browser. If the user agent string (shown in Figure 9.2) indicates that the client is an iPhone or iPod user, then the **On-Demand Cert Auth - Require** authentication mode is executed. Otherwise, **On-Demand Cert Auth - Request** is executed.



**Figure 9.1** On-Demand authentication agent example



*Figure 9.2 iPhone/iPod user string example*

## Adding the On-Demand Cert Auth agent to your access policy

After you create a **clientssl** profile with Client Certificate set to **ignore**, you can add an On-Demand certificate authentication agent to your access policy. This action requires that the client has a valid certificate on its machine before it runs the certificate authentication. F5 Networks highly recommends that a Decision Box agent precede the On-Demand certificate authentication agent in the visual policy editor so that the user has the option of indicating whether he has a valid certificate. If a valid certificate is not available, and indicated as such in the Decision Box agent, the system bypasses the client certificate validation process and proceeds to the next step in the verification process.

### ◆ Note

*If you want to authenticate the client with a valid certificate at the beginning of the initial SSL handshake of your access policy, do not use the On-Demand Cert Auth agent.*

### To add an On-Demand certificate authentication agent to an access policy

1. Select an access policy or create a new one.
2. On the navigation pane, expand **Access Policy**, and select **Access Profiles**.  
The Access Profile screen opens.
3. Click the access policy and select **Edit**.  
The visual policy editor screen opens.
4. Under Predefined Actions, and in the **Authentication** settings, select **On-Demand Cert Auth**.
5. Click **Add Item**.  
A Properties screen opens.

6. From the **Auth Mode** option, select either **Request** or **Required**. The default is **Request**.

***Configuring AUTH MODE for use with iPod and iPhone***

*Select **Required**. Note that to pass a certificate check using Safari, you will be asked to select the certificate multiple times. This is expected behavior.*

7. Click **Save**.  
The system adds the On-Demand Certificate authentication agent to your access policy.

---

**◆ Note**

*If your access policy is configured with an On-Demand certificate authentication action, the user's browser must have a valid certificate. Otherwise, your browser may stop responding because the client failed to provide a valid certificate. To avoid running into this problem, we highly recommend that you use the Decision box agent in your access policy so that the users are given an option to specify whether or not they have a valid certificate.*

## Configuring client SSL profiles

The BIG-IP® system provides a simple way to configure your client SSL profile so that you can include the cerClient certificate intificate authentication process in your access policy.

To ensure that your client profile is set up correctly, you must perform these tasks, sequentially.

- Importing a certificate and the corresponding key
- Configuring the **clientssl** profile
- Adding an On-Demand Certificate agent into your access policy

### Importing a certificate and the corresponding key

The first task in configuring a client SSL profile is to import a certificate and the corresponding key (issued by your organization CA).

#### To import a certificate and a key

1. In the navigation pane, expand **Local Traffic** and click **SSL Certificates**.  
The SSL Cert screen opens.
2. Click the **Import** button.  
The SSL Certificate/Key Source screen opens.
3. Select an **Import Type** from the list, type the required parameters into the boxes, and click the **Import** button.  
The screen refreshes to show settings specific to the type you selected.

### Configuring a clientssl profile

The next task is to configure a **clientssl** profile.

#### To configure the clientssl profile

1. In the navigation pane, expand **Local Traffic** and click **Profiles**.  
The HTTP Profiles screen opens.
2. From the SSL menu, choose Client.  
The Client SSL Profiles screen opens.
3. At the upper right, click **Create**.  
A New Client SSL Profile screen opens.
4. In the **Name** box, type a name for your **clientssl** profile.
5. In **Configuration**, select **Advanced** from the list.
6. Check the **Custom** box.



7. For the **Trusted Certificate Authorities** setting, select your trusted certificate authority.
8. For the **Ciphers** setting, type in a NATIVE cipher to support the On-Demand certificate authentication check. The list of supported NATIVE ciphers includes the following:
  - RC4-MD5
  - RC4-SHA
  - AES128-SHA
  - AES256-SHA
  - DES-CBC3-SHA
  - DES-CBC-SHA
  - EXP1024-RC4-MD5
  - EXP1024-RC4-SHA
  - EXP1024-DES-CBC-SHA
  - EXP-RC4-MD5
  - EXP-DES-CBC-SHA
  - NULL-MD5
  - NULL-SHA
9. In the Client Authentication area, check the **Custom** box. Your choice for the **Client Certificate** setting depends on the type of agent you want to use in your access policy. To use:
  - On-Demand certificate authentication agent: Select **ignore**.
  - Client certificate inspection agent:

For iPod or iPhone clients, select **require**.

For all other clients, we recommend that you select **request**.
10. Click **Finished**.

Your **clientssl** profile is now created.

## Using certificates to authenticate users

There tasks are required for using either the On-Demand Cert Auth or Client Cert Inspection action:

- Install the client root certificate on the Access Policy Manager.
- Instruct users how to download and install the certificate on their computers. You can also email the certificate to users.

## Understanding certificate revocation status

Access Policy Manager supports three ways to validate certificate revocation status:

- CRLs
- OCSP
- CRLDP

For more detailed information, refer to **BIG-IP® Local Traffic Manager™: Concepts** available on the AskF5™ web site at <http://support.f5.com>.

## Understanding CRLs

A **certificate revocation list** (CRL) is a list of revoked (invalid) certificates. The CRL describes the reason for the revoked status of the certificate, and provides the certificate's issue date and originator. The list also notes its next update.

When a user with a revoked SSL certificate attempts to log on to the Access Policy Manager, the system allows or denies access based on the CRL configured in the profile.

A CRL is one of three common methods for maintaining valid, certificate-based access to servers in a network. **CRLDP** is an industry-standard protocol designed to manage SSL certificates revocation on a network or system. The main limitation of CRL is that the current state of the CRL requires frequent updates. Whereas, OCSP checks certificate status in real time. You can read more about OCSP in *Understanding OCSP*, following.

The CRL is a PEM-formatted file containing a list of revoked certificate attached to the client SSL profile. Make sure the CRL file is kept up-to-date. You must manually install the CRL file to the **/config/ssl/ssl.crl** directory since this is not an automatic process.

### To attach a PEM-formatted CRL file to a client SSL profile.

1. In the navigation pane, expand **Local Traffic**, and click **Profiles**. The HTTP Profiles screen opens.
2. From the SSL menu, choose Client. The Client SSL Profiles screen opens.
3. Click the name of a profile to open a screen where you can edit it.
4. In the Client Authentication area, select the **Custom** checkbox to enable editing.
5. From the **Certification Revocation List (CRL)** box, select the name of the CRL file that you previously imported into **/config/ssl/ssl.crl/** on the BIG-IP system.
6. Click **Update**. Your CRL file is now attached to the **client SSL** profile.

Note that if you have multiple CRL files, you cannot aggregate them into one master file. You must point to the individual file (in PEM format) if you want to retrieve CRL information.

◆ **Note**

*You should not configure CRL updates if you are using the BIG-IP system to generate and issue SSL certificates to users (using either a self-signed client root CA certificate, or a client root CA certificate from a trusted CA). In this case the Access Policy Manager manages CRLs internally.*

## Converting DER files to PEM file format

The Access Policy Manager system supports CRL files only in PEM format. However, you can convert non-PEM file format, such as DER, by using a few CLI commands.

### To convert a DER file to PEM format

1. Use SSH to access the Access Policy Manager system.
2. Run the command **crl -inform DEM -outform PEM -in CRL.crl -out CRL.PEM**.

You have successfully converted your input CRL file, **CRL.crl** in DER format to output CRL file, **CRLpem** in PEM format.

## Understanding OCSP

The *Online Certificate Status Protocol (OCSP)* enables applications to determine the revocation status of a certificate. OCSP provides more timely revocation information than is possible using CRLs, and may also be used to obtain additional status information. An OCSP client, in this case the Access Policy Manager, 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.

The Access Policy Manager supports OCSP validation of SSL certificates.

◆ **Note**

*Do not use OCSP if you are using the BIG-IP system to generate/issue SSL certificates to users (using either a self-signed client root CA certificate, or a client root CA certificate issued by a trusted CA). In this case, the Access Policy Manager is managing CRLs internally.*

Setting up OCSP requires these tasks:

- Configuring an OCSP responder object
- Creating an SSL OCSP profile
- Binding the SSL OCSP profile to a virtual server

## Configuring an OCSP responder object

To work with OCSP, you first must create an OCSP responder object.

### To configure an OCSP responder object

1. In the navigation pane, expand **Local Traffic**, and click **Profiles**.  
The HTTP Profiles screen opens.
2. From the Authentication menu, choose OCSP Responders.  
The OCSP Responders screen opens.
3. At the upper right, click **Create**.  
A General Properties screen opens.
4. Type in a name for your OCSP profile. The name should not contain capital letters (which generates an error).  
This screen refreshes to display additional parameters specific to your selection.
5. In the **URL** setting, type the URL for your external OCSP responder.  
A separate OCSP responder object must be created for each OCSP server.
6. Specify a **Certificate Authority File**.
7. Click **Finished**.

## Creating an SSL OCSP profile

You must create an SSL OCSP profile in order for OCSP to work properly.

### To create an SSL OCSP profile

1. In the navigation pane, expand **Local Traffic**, and click **Profiles**.  
The HTTP Profiles screen opens.
2. From the Authentication menu, choose Profiles.  
The Authentication Profiles screen opens.
3. At the upper right, click **Create**.  
The New Authentication Profile screen opens.
4. Type in a name for your OCSP profile server, and select SSL OCSP from the **Type** list.
5. Click **Finished**.  
This creates the SSL OCSP profile.

## Binding the SSL OCSP profile to a virtual server

The last step in setting up OCSP is to include the created OCSP profile in the authentication profile settings of the virtual server.

### To bind the OCSP to a virtual server

1. In the navigation pane, expand **Local Traffic**, and click **Virtual Servers**.  
The General Properties screen opens.
2. From the **Configuration** list, select **Advanced**.
3. For **Authentication Profiles settings**, from the **Available** box, select the SSL OCSP profile you want to bind to the virtual server.
4. Click the move button (<<) to move the SSL OCSP profile to the **Enabled** box.
5. Click **Update**.

## Using CRLDP

**CRLDP** stands for Certificate Revocation List Distribution Point. CRLDP checks the revocation status of an SSL certificate as part of authenticating that certificate. CRL distribution points are used to distribute certificate revocation information across a network. A **distribution point** is a URI or directory name specified in an SSL certificate that identifies how the server obtains CRL information. In addition, distribution points can be used in conjunction with CRLs to configure certificate authorization using any number of LDAP servers.

In setting up CRLDP, you complete the following tasks:

- Configuring a CRLDP server object
- Configuring a CRDLP configuration object
- Creating a CRLDP profile
- Binding the CRLDP profile to a virtual server.

### Configuring a CRLDP server object

When you set up a CRLDP server object, you include details such as the CRLDP server IP address, a port for the CRLDP authentication traffic, and the LDAP base DN for certificates that specify the CRL distribution point in **directory** name format. The base DN is used when the value of the X.509 v3 attribute CRLDP is of type **dirName**. In this case, the Access Policy Manager attempts to match the value of the CRLDP attribute to the base DN value.

#### To configure a CRLDP Server object

1. In the navigation pane, expand **Local Traffic**, and click **Profiles**.  
The HTTP Profiles screen opens.
2. From the Authentication menu, choose CRLDP Responders.  
The CRLDP Responders screen opens.
3. At the upper right, click **Create**.  
The General Properties screen opens.
4. Fill in all the details for this screen  
Refer to the online help for specific details on each settings.
5. Click **Finished**.  
This creates a CRLDP Server object.

### Configuring a CRLDP configuration object

When you configure a CRLDP configuration object, you include details about the CRLDP servers that allow you to use the certificate issuer to extract the CRLDP.

### To configure a CRLDP configuration object

1. In the navigation pane, expand **Local Traffic**, and click **Profiles**.  
The HTTP Profiles screen opens.
1. From the Authentication menu, choose Configurations.  
The Authentication Configurations screen opens.
2. At the upper right, click **Create**.  
The General Properties screen opens.
3. In the **Name** box, type in a name for your CRLDP configuration object.
4. From the **Type** list, select **CRDLP**.  
Additional configuration parameters appear.
5. Specify your CRLDP server and click **Finished**.  
This creates the CRLDP configuration object.

## Creating a CRLDP profile

To use CRDLP, you must create a CRLDP profile and reference the CRLDP configuration object.

### To create a CRLDP profile

1. In the navigation pane, expand **Local Traffic**, and click **Profiles**.  
The HTTP Profiles screen opens
2. From the Authentication menu, choose Profiles.  
The Authentication Profiles screen opens.
3. At the upper right, click **Create**.
4. In the **Name** box, type in a name for your CRLDP profile.
5. From the **Type** list, select **CRLDP**.  
Additional configuration parameters become available.
6. Enable all the **custom** check boxes and configure all settings.  
Refer to the online help for specific details on each settings.
7. Click **Finished**.  
This creates the CRLDP profile.

## Binding the CRLDP profile to a virtual server

The last step in setting up CRDLP is to include the CRLDP profile in the authentication profile settings of the virtual server.

### To bind the CRDLP profile to a virtual server

1. In the navigation pane, expand **Local Traffic**, and click **Virtual Servers**.  
The Virtual Server List screen opens.



2. From the list of virtual servers, click the name of the server you want to bind the CRDLP profile.  
The Properties screen opens.

3. From the **Configuration** setting, select **Advanced**.

From the **Available** box, for the **Authentication Profiles**, select the CRDLP profile you want to bind to the virtual server.

4. Click the move button (<<) to move the SSL OCSP profile to the **Enabled** box.
5. Click **Update**.  
The CRDLP Profile is now associated with your virtual server.





# 10

---

## Configuring Virtual Servers

---

- Introducing virtual servers with Access Policy Manager
- Configuring virtual servers for access policies
- Configuring a local traffic virtual server with an access policy



## Introducing virtual servers with Access Policy Manager

With BIG-IP® Access Policy Manager®, you configure virtual servers with particular configurations for access policies. For web access management, you configure an existing Local Traffic Manager™ virtual server to use an access policy, or you can create a new virtual server for this purpose. The IP address assigned to a virtual server is the one that is typically exposed to the Internet for SSL VPN services.

When creating a virtual server, specify that the virtual server is a host virtual server for Access Policy Manager, and not a network virtual server. (For more information on host and network virtual servers, see the *Configuring Virtual Servers* chapter in the **BIG-IP® Local Traffic Manager™: Concepts** guide available on the AskF5™ web site at <http://support.f5.com>. In either case, you need only configure a few settings: a unique name for the virtual server, a destination address, and a service port.

---

### ◆ Important

*When you create a virtual server, the BIG-IP system places the virtual server into your current administrative partition. For information on partitions, see the **TMOS® Management Guide for BIG-IP® Systems**.*

For production deployment of your configuration, you should either edit the **clientssl** profile to use your imported certificate and key, or create a new profile based on the **clientssl** profile that uses your own certificate and key. For more information, see *Configuring a clientssl profile*, on page 9-8. For initial evaluation of Access Policy Manager, you may select the default **clientssl** profile in the SSL Profile (Client) list. This default profile does not contain a valid SSL server certificate, but it can be used for initial Access Policy Manager evaluation and testing.

## Understanding SNAT interactions

The following interactions apply to SNAT settings with access policies.

- The SNAT setting for a network access tunnel is applied in the Network Access resource's Advanced Network Settings.
- The SNAT settings for App Tunnels, Optimized Applications, Remote Desktops, Portal Access, Citrix connections, and web access management deployments, are applied in the access policy session, if configured. If there is no specific SNAT configuration specified, the SNAT settings from the virtual server are applied.
- To configure SNAT settings for an access policy session, add a Route Domain Selection access policy item.
- If the Access Policy Manager traffic hits another user defined virtual server before leaving the BIG-IP® system, the SNAT settings from the last user defined virtual server are used on outgoing connections.

## Configuring virtual servers for access policies

You create a virtual server to provide a portal for user logons to Access Policy Manager resources. At a minimum, you must create one virtual server on which your users can log on.

### To create a virtual server for a secure connection

1. On the Main tab of the navigation pane, expand **Local Traffic**, and click **Virtual Servers**.  
The Virtual Server List screen opens.
2. Click **Create**.  
The New Virtual Server screen opens.
3. In the **Name** box, type a name for the virtual server.
4. In the **Destination** area, select **host**.
5. In the **Address** box, type the virtual server host IP address.
6. From the **Service Port** list, select **HTTPS**.
7. From the **HTTP Profile** list, select **http**.
8. From the **SSL Profile (Client)** list, select the client SSL profile to use with this virtual server.
9. If your web application server is using HTTPS services, from the **SSL Profile (Server)** list, select the server SSL profile to use with this virtual server.
10. For a portal access virtual server, from the **SNAT Pool** list, select **Auto Map**.  
*See Understanding SNAT interactions, on page 10-1 for a warning about the SNAT Pool setting.*
11. If you are configuring a virtual server that will forward traffic to another server or is forwarded to by another server, from the **Source Port** list, select **Change**.  
*This option only appears when you select **Advanced** for the Configuration section.*
12. From the **Access Profile** list, select the access profile to associate with this virtual server.  
You must create this access profile before you define the virtual server. There is no default access profile available.
13. From the **Connectivity Profile** list, select the connectivity profile to associate with this virtual server.  
There is no default connectivity profile, so you must create a connectivity profile before you can select one from this list.
14. If you are creating a virtual server to use with portal access, from the **Rewrite Profile** list, select the rewrite profile.  
You can select a rewrite profile with a network access or application access configuration.

15. If you are creating a virtual server to use with portal access in minimal patching mode, from the **Default pool** list, select the local traffic pool for this application.
16. Click **Finished** to complete the configuration.

## Creating a virtual server for DTLS

To configure DTLS mode for a network access connection, you must configure a virtual server specifically for use with DTLS. This DTLS virtual server must have the same IP address as the TCP (HTTPS) virtual server to which a user connects to start an Access Policy Manager session. The network access resource assigned by the access policy on the TCP virtual server sharing the same address must be configured with the DTLS option selected. After the Access Policy Manager session is established, the network access tunnel is started using the DTLS virtual server, on the same IP address.

For more information, see the ***BIG-IP® Access Policy Manager® Network Access Guide***.

### To create a virtual server for use with DTLS

1. On the Main tab of the navigation pane, expand **Local Traffic**, and click **Virtual Servers**.  
The Virtual Server List screen opens.
2. Click **Create**.  
The New Virtual Server screen opens.
3. In the **Name** box, type a name for the virtual server.
4. In the **Destination** area, select **host** for the type of virtual server.
5. In the **Address** box, type the virtual server host IP address.  
This is the same IP address as the TCP virtual server to which your users connect.
6. In the **Service Port** box, type the port number that you specified in the Network Access resource configuration, in the **DTLS Port** box. By default, the DTLS port is **4433**.
7. In the Configuration area, from the **Protocol** list, select **UDP**.
8. If you are configuring a virtual server that will forward traffic to another server or is forwarded to by another server, from the **Source Port** list, select **Change**.  
*This option only appears when you select **Advanced** for the Configuration section.*
9. From the **Connectivity Profile** list, select the connectivity profile associated with this virtual server.  
This profile specifies client connection behavior and configuration.
10. From the **SSL Profile (Client)** list, select the client SSL profile to use with this virtual server.

*The system automatically uses DTLS hardware acceleration, if supported by the hardware. To set the system to disable DTLS hardware acceleration, see **Configuring a client SSL profile to disable DTLS acceleration**, on page 10-4.*

11. Click **Finished** to complete the configuration.

### **Configuring a client SSL profile to disable DTLS acceleration**

1. On the Main tab of the navigation pane, expand **Local Traffic**, and click **Profiles > SSL > Client**.  
The SSL Client Profiles screen opens.
2. Click the SSL client profile you want to edit.  
The Client SSL Profile Properties screen appears.
3. Next to **Configuration**, select **Advanced**.
4. Click the **Custom** check box.
5. In the Ciphers box, type **DEFAULT:RFCDTLS**.
6. Configure the other fields in the client SSL profile as required for your configuration.
7. Click **Update**.



## Configuring a local traffic virtual server with an access policy

To configure virtual servers for web access management, you must configure both the BIG-IP® Local Traffic Manager™ and Access Policy Manager®.

When you configure for this method of access, you create a virtual server that has one or more pool members and HTTP servers, and you attach an access policy to that virtual server. For more details, see Chapter 2, .

### To create a virtual server for web access management

1. On the Main tab of the navigation pane, expand **Local Traffic**, and click **Virtual Servers**.
2. Click **Create**.
3. Type the name and address of the virtual server.
4. Select a service port.
5. Select the **HTTP Profile** from the available options.  
The default profile, **http**, is usually sufficient, unless additional configuration options are needed.
6. Select the **SSL profile (Client)** setting.  
A client SSL profile is only required if you want to enable SSL from the client to the virtual server.
7. Select the **SSL profile (Server)** setting.  
A server SSL profile is only required if the pool members require SSL.
8. If you are configuring a virtual server that will forward traffic to another server or is forwarded to by another server, from the **Source Port** list, select **Change**.  
*This option only appears when you select **Advanced** for the Configuration section.*
9. From the **Access Profile** list, select an access profile you created for web access management.
10. Click **Finished**.
11. The Virtual Server List screen opens.
12. Click the Resources tab.
13. From the Default Pool list, select a default pool.  
To configure and create local traffic pools, see **BIG-IP® Local Traffic Manager™: Concepts**.
14. Click **Update**.





II

---

## Advanced Topics in Access Policies

---

- Setting up a logon page to collect user credentials
- Example: Using a customized logon page to collect user credentials
- Using multiple authentication methods
- Example: Using client certificate authentication with Active Directory
- Configuring policy routing
- Example: Directing users to different route domains
- Using advanced access policy rules



## Setting up a logon page to collect user credentials

In most applications, a logon page is used to present user name and password prompts to a user, to collect the credentials the user enters, and to forward those credentials on to an authentication method. In BIG-IP® Access Policy Manager®, you use the visual policy editor to assign a logon page in an access policy. This section describes the logon action, and how to customize the page presented by the logon action.

### Understanding the logon page action

The logon page customization elements include the information that appears between the header and the footer. You customize this information using the Logon Page action in the access policy configuration. The default English logon page configuration appears in Figure 11.1.

Logon Page Agent				
	Type	Post Variable Name	Session Variable Name	Read Only
1	text	username	username	No
2	password	password	password	No
3	none	field3	field3	No
4	none	field4	field4	No
5	none	field5	field5	No


  

Customization	
Language	en
Form Header Text	Secure Logon   for F5 Networks
Logon Page Input Field #1	Username
Logon Page Input Field #2	Password
Logon Page Input Field #3	Field 3
Logon Page Input Field #4	Field 4
Logon Page Input Field #5	Field 5

*Figure 11.1 Default logon page action configuration*

### To customize the logon page action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.

3. On a branch of the access policy, click the plus sign (  ) to add an action.

The Add Item popup screen opens.

4. On the Logon tab, select **Logon Page** and click **Add Item**.  
The **Logon Page** configuration popup screen opens.

5. Select the language you want to customize.

6. Customize the logon page agents:

For each **Logon Page Agent** you are using, customize the type of logon page agent. For each agent you can specify a **Post Variable Name**, **Session Variable Name**, and whether the agent is **Read Only**. See *Adding and customizing a logon page*, on page 6-1, for more information.

7. Customize the elements in the Customization section.

- **Form Header Text** - Specifies the text that appears at the top of the login box.
- **Logon Page Input Field # (1-5)** - These fields specify the text that is displayed on the logon page for each of the logon page agents, defined in the Logon Page Agent screen area.
- **Save Password Checkbox**- Specifies the text that appears adjacent to the check box that allows users to save their passwords in the logon form. This field is used only in the secure access client, and not in the web client.
- **Logon Button** - Specifies the text that appears on the logon button, which a user clicks to post the defined logon agents.
- **Front Image** - Specifies an image file to display on the logon page.  
Click **Browse** to select a file from the file system. Click **Show image** or **Hide Image** to show or hide the currently selected image file. Click **Revert to Default Image** to discard any customization and use the default logon page image.
- **New Password Prompt** - Specifies the prompt displayed when a new Active Directory password is requested.
- **Verify Password Prompt** - Specifies the prompt displayed to confirm the new password when a new Active Directory password is requested.
- **Password and Password Verification do not Match** - Specifies the prompt displayed to confirm the new password when a new Active Directory password is requested.

8. Click **Save** when the settings are customized.

## Example: Using a customized logon page to collect user credentials

In this example, a logon page action is added to an access policy. The logon page action presents the logon information to a user who attempts to start a network access connection. In this example, the English language logon page is customized with several fields for the fictitious company Bogon Networks, Inc. In addition, the user name, password, and logon fields are customized, and the footer message is changed.

### To add a logon page action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. Click **Create**.  
The New Access Profile screen opens.
3. In the **Name** box, type **BogonNet1**, then click **Finished**.  
The Access Profile Properties screen opens.
4. Click the Access Policy tab, then click **Edit Access Policy for Profile “BogonNet1”**.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
5. Click the plus sign ( **+** ) to add an action.  
The Add Item popup screen opens.
6. On the Logon tab, select **Logon Page** and click **Add Item**.  
The Logon Page action popup screen opens.
7. From the **Language** list, select **en** to customize the logon page for English.
8. In the **Form Header Text** box, type **Secure Logon <br> for Bogon Networks, Inc.**
9. In the **Logon Page Input Field #1** box, type **User ID:**.
10. In the **Logon Page Input Field #2** box, type **Passcode:**.
11. In the **Logon Button** box, type **LOGON**.  
The final configuration is shown in Figure 11.2, following.
12. Click **Save**.
13. Click **Apply Access Policy**.
14. Close the browser tab or window and return to the Access Policy screen.



Form Header Text	Secure Logon   for Bogon Networks, Inc.
Logon Page Input Field #1	User ID:
Logon Page Input Field #2	Passcode:
Logon Page Input Field #3	Field 3
Logon Page Input Field #4	Field 4
Logon Page Input Field #5	Field 5
Logon Button	LOGON
Front Image	
Save Password Checkbox	Save Passcode

*Figure 11.2 Logon Page action customization example popup screen*

### To customize the logon page footer

#### ◆ Note

Typically you configure the logon page by adding your own custom logo and graphics. To simplify this example, the header box is left as the default with the F5 graphics and background color.

1. On the Access Policy screen, click the Customization tab.
2. From the **Customization Type** list, select **general UI**.
3. From the **Language** list, select **en**.
4. Click **Find Customization**.

5. Under Page Footer Settings, in the **Footer Text** box, type **For use by employees of Bogon Networks, Inc., and subsidiaries.<br>Copyright © 2009 Bogon Networks, Inc.<br>All rights reserved.**
6. Click **Update**.
7. Click **Apply Access Policy**.

## Using multiple authentication methods

In an access policy you can use multiple authentication methods by adding multiple authentication actions. With multiple authentication methods, you can add two-factor authentication to your access policy. You can also use multiple authentication methods to assign different resources or routes depending on the authentication method.

### Client certificate two-factor authentication

You can use two or more authentication methods in an access policy. This example uses a client certificate for authentication, followed by Microsoft Active Directory authentication. The Active Directory action uses the authentication information collected in the logon page action that precedes it. After the user is authenticated, the access policy assigns resources with the resource assign action, and the user is allowed access.

The configuration for this access policy is described in the section following.

## Example: Using client certificate authentication with Active Directory

In this example, a user who logs on to the network must have both a valid client certificate, and an account on the Microsoft Active Directory server. The following shows the sequence of events that occur in this example.

- The access policy first verifies the user's operating system is Windows 7, Windows 8, or Windows XP. This step is optional.
- The user's client certificate is trusted.
- If the user's certificate check action passes successfully, the user sees a logon page. If the user's certificate action does not pass successfully, the user sees a logon denied page.
- On the logon page, the user inputs credentials, and the access policy tests these credentials against Active Directory.
- If the Active Directory check succeeds, the Access Policy Manager assigns resources to the user, and the user is assigned a connection and can begin working with network resources. The user also sees a webtop, if one is assigned.

## Configuring the client certificate two factor authentication with Active Directory example

This example provides a guide to the tasks involved in the configuration of this access policy. Note that this is not a step-by-step procedure, but a list of procedures, with references to the tasks that you must perform to complete the example.

### To configure the access policy

1. (Optional) Add the Client OS action.  
See *Setting up the client OS check*, on page 8-2. Configure the Client OS access policy item with one rule that specifies the Client OS is Windows 7, Windows 8, or Windows XP. Delete the other rules. You can optionally rename the Client OS access policy item.
2. Add the Client Cert Auth action on the successful rule branch of the access policy.  
See *Adding the On-Demand Cert Auth agent to your access policy*, on page 9-6.
3. Add the Logon Page action to the successful rule branch of the access policy.  
See *Adding and customizing a logon page*, on page 6-1.
4. Add the Active Directory auth action to the successful rule branch of the access policy.  
See the **BIG-IP® Access Policy Manager® Authentication Configuration Guide**.

5. Add the resource assign action to the successful rule branch of the access policy.  
The resource assign action must set a network access resource. You can optionally assign ACLs, and a network access webtop. See *Assigning resources*, on page 6-8.
6. Change the ending of the successful branch of the access policy to an Allowed ending.  
See *Using policy endings*, on page 5-10.
7. Click **Apply Access Policy** to start the access policy.

## Configuring policy routing

You can use policy routing in a number of different scenarios to provide users access to different network segments or resources. For example, you might create a route domain that connects unauthenticated users on a publicly available wireless segment only to the external web, while denying access to internal network resources. To create this configuration, you can use a route domain and SNAT selection action in the access policy on the fallback rule branch of an authentication action, to send failed logons to a separate route domain from the internal network.

Access Policy Manager® uses route domain objects to provide access to routing features in access policies. The BIG-IP® system supports the ability to configure multiple route domains. A *route domain* is a BIG-IP system object that represents a particular network configuration. After creating a route domain, you can associate various BIG-IP® system objects with the domain: unique VLANs, routing table entries such as a default gateway and static routes, self IP addresses, virtual servers, and pool members.

Route domains provide the capability to segment network traffic, and define separate routing paths for different network objects and applications. Because route domains segment the network traffic, they also provide the capability to have separate IP networks on the same unit, where each route domain uses the same IPv4 address space. Using routing domains, you can assign the same IP address or subnet to more than one device on a network, as long as each instance of the IP address resides in a separate routing domain.

To configure policy routing, you must configure a route domain. For more information on configuring route domains, see the *TMOS® Management Guide for BIG-IP® Systems*.

## Setting up route domain selection in an access policy

Once you have defined a route domain, you can route users to the route domain in the access policy, using the route domain selection action.

### To add a route domain selection action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. On a rule branch of the access policy, click the plus sign ( **+** ) to add an action.  
The Add Item popup screen opens.

4. On the Assignment tab, select **Route Domain and SNAT Selection** and click **Add Item** to add the action to the access policy.  
The Route Domain and SNAT Selection action popup screen opens.
5. From the **Route Domain ID** list, select the route domain ID.
6. Click **Save** to complete the configuration.

## Example: Directing users to different route domains

In this example, your company has switched from RADIUS authentication to Active Directory authentication, but has not yet completed the full transition. Because of the state of the authentication changeover, you would like your legacy RADIUS users to pass through to a portal access connection on a separate router, instead of allowing full access to your network.

This example requires you to configure:

- A route domain.
- An access profile.
- An access policy that contains a logon page, an Active Directory Authentication action, a RADIUS authentication action, two resource assign actions, and a route domain and SNAT selection action.

## Configuring the policy routing example

To configure this example, you must define a route domain and create an access policy that references that route domain. To keep the access policy generic enough for any implementation, the example does not specify names or addresses for the Active Directory server or the RADIUS server to use with the authentication action. The example also does not specify the portal access or network access resources to use with the resource assign actions. You can create the access policy without configuring these actions, and add your own servers and resources.

### To configure the route domain

1. On the Main tab of the navigation pane, expand **Network**, and click **Route Domains**.  
The Route Domain List screen appears.
2. Click the **Create** button.  
The New Route Domain screen opens.
3. In the **ID** box, type **1** for the ID for the new route domain.
4. In the VLANs section, from the **Available** list, select an available VLAN and click the << button to move the VLAN to the Members list.
5. Click **Finished**.

### To create the routing access profile




1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. Click the **Create** button.  
The New Profile screen opens.






3. In the **Name** box, type a name for the access profile, for example, **PolicyRouteTest**.
4. Click **Finished**.  
The Access Policy screen appears.

Continue on to configure the access policy.

### To configure the access policy

1. On the access policy screen, click the link, **Edit Access Policy for Profile**.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
2. On the fallback branch of the access policy, click the plus sign (  ) to add an action.  
The Add Item popup screen opens.
3. On the Logon tab, select the **Logon Page** action, and click **Add Item**.  
The Logon Page action popup screen opens.
4. Click **Save** to save and close the action.
5. Click the plus sign (  ) on the fallback branch after the logon page action.  
The Add Item popup screen opens.
6. On the Authentication tab, select **AD Auth**, and click **Add Item**.  
The Active Directory Authentication action popup screen opens.
7. From the **Server** list, select an Active Directory server.  
If you do not have an Active Directory server, you can leave the action unconfigured for the purposes of the example.
8. Click **Save** to save the action.
9. On the successful branch following the Active Directory action, click the plus sign (  ) to add an action.  
The Add Item popup screen opens.
10. On the Assignment tab, select the **Resource Assign** action, and click **Add Item**.  
The Resource Assign action popup screen opens.
11. Click the **Add new entry** button.
12. Click the **Set Network Access Resource** link, select a network access resource to assign to clients who successfully authenticate with Active Directory, and click the **Update** button.
13. Optionally, click the **Set Webtop** link, and select a network access webtop to assign to clients who successfully authenticate with Active Directory, then click the **Update** button.
14. Click **Save** to save the action.

15. On the fallback branch following the Active Directory action, click the plus sign (  ) to add an action.  
The Add Item popup screen opens.
16. On the Authentication tab, select the **RADIUS Auth** action, and click **Add Item**.  
The RADIUS authentication action popup screen opens.
17. From the **AAA Server** list, select a RADIUS server.  
If you do not have a RADIUS server, you can leave the action unconfigured for the purposes of the example.
18. Click **Save** to save the action.
19. On the successful branch following the RADIUS action, click the plus sign (  ) to add an action.  
The Add Item popup screen opens.
20. On the Assignment tab, select the **Route Domain and SNAT Selection** action, and click **Add Item**.  
The Route Domain Selection action popup screen opens.
21. From the **Route Domain ID** list, select **1**.  
This assigns the route domain gateway you defined earlier to clients who successfully authenticate to the RADIUS server.
22. Click **Save** to save the action.
23. On the successful branch following the route domain selection action, click the plus sign (  ) to add an action.  
The Add Item popup screen opens.
24. On the Assignment tab, select the **Advanced Resource Assign** action, and click **Add Item**.  
The Resource Assignment action popup screen opens.
25. Click the **Add new entry** button.  
A new row displays in the table.
26. Click the **Add/Delete** link in the new row.  
An additional popup screen opens.
27. On the Network Access tab, select a network access resource to assign to clients that successfully authenticate with RADIUS.
28. Optionally, on the Webtop tab, select a network access webtop to assign to clients who successfully authenticate with Active Directory.  
*Note that you can assign the same network access resource to both types of clients, and because a different route domain is specified in the route domain selection action, the clients will still reach separate routers.*
29. Click the **Update** button.  
The popup screen closes.
30. Click **Save** to save the action.  
The popup screen closes.

31. Click the ending following the resource assign actions and change it to an allow ending, by selecting **Allow** and clicking **Save**.

## Using advanced access policy rules

You can use advanced rules in an access policy to provide customized functionality to users. This functionality is useful when the default access policy rules and the rules created with the expression builder do not provide functionality you require.

When you write an expression in the Advanced tab of the rule popup screen, a non-zero return value typically causes the rule to be evaluated as true or successful, and the access policy follows the corresponding rule branch. The return value of **0** causes the rule to be evaluated as false, and the rule follows the corresponding branch, or a fallback branch.

## Understanding advanced access policy rule situations

You can use advanced access policy rules in four situations in the visual policy editor.

- ◆ You can use an advanced access policy rule to make flexible decisions after an access policy action completes. To do this, you add the advanced access policy rule on the Advanced tab in the Expression popup screen of an action.

In this scenario, if the value returned by the expression is not zero, the rule is evaluated as true, and the access policy runs and follows the corresponding rule branch. If the value returned by the expression is zero, the rule is evaluated as false, and the access policy follows the branch assigned to the negative response (typically a fallback branch).

- ◆ You can use an advanced access policy rule to add flexibility when assigning resources to users. To do this, you add the advanced access policy rule on the Advanced tab in the Expression popup screen of the resource assign action.

In this scenario, if the value returned by the expression is not zero, the resource assignment rule is evaluated true, and the corresponding resource or ACL is assigned to the user. If the value returned by the expression is zero, the resource assignment rule is evaluated as false, and the resource or ACL is not assigned.

- ◆ You can use an advanced access policy rule to add flexibility by creating a custom session variable, and then assigning the session variable in other advanced access policy rules. To do this, you use the custom variable and custom expression options in the variable assign action.

In this scenario, the value returned by the custom expression is assigned to the custom variable.

- ◆ You can use an advanced access policy rule to override the properties of an assigned network access resource. To do this, you assign a configuration variable to a custom expression, in the variable assign action.

In this scenario, the value returned by the expression is used to overwrite the value of the selected property from the network access resource.

## Writing advanced access policy rules

Advanced access policy rules are written in the Tcl programming language. An advanced access policy rule is a Tcl program. You can use the various facilities provided by the Tcl language in advanced access policy rules. For example, you can use loops (**while**, **foreach**, and so on), conditions (**ifelse**, **switch**, and more), functions (**proc**), and built-in Tcl commands (**strings**, **split**, for instance) as well as various Tcl operators.

For comprehensive documentation on the Tcl language, see <http://www.tcl.tk/doc/>.

## Understanding the mcget command

In Access Policy Manager access policies, session variables are accessed from system memory during the evaluation of an access policy rule. Access Policy Manager stores all session variables generated in a session in its memory cache. The Tcl command that gets these variables is **mcget**, which is an abbreviation for “**get** the session variable from the **memory** cache.”

The general syntax to access a session variable follows.

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

In this example, the name of the session variable, **session.ssl.cert.cn**, is enclosed in braces { }. The brackets [ ] that enclose the entire command are the TCL notation for command evaluation.

## Using a Tcl expression or program as an advanced access policy rule

You can use a Tcl expression or a complete Tcl program as an advanced access policy rule. The return value of the expression or program is used to evaluate the access policy rule. For example, the following access policy rule uses a TCL expression to check if the Organizational Unit (OU) field of a user certificate contains the text **PD**.

```
expr { [mcget {session.ssl.cert.OU}] contains "PD" }
```

The return value of the expression is the return value used in the access policy rule.

---

### ◆ Note

*The Tcl language specifies that the expression begin with the syntax **expr**. For a complete description of the various operators and syntax allowed in a Tcl expression, see <http://www.tcl.tk/man/tcl8.0/TclCmd/expr.htm>.*

## Understanding advanced access policy rule limitations

In Access Policy Manager, the Tcl code entered in an action is not validated for proper Tcl syntax. If there is a Tcl syntax error in a rule, this error is not caught at configuration time, but the rule fails at session establishment time. We recommend that you test rules with an independent Tcl shell before they are configured in the access policy to avoid this.

The semicolon separator (;) is required between two consecutive Tcl statements. This is not the same as using the default newline (\n) as a separator.

---

### ◆ Note

*The name space for Access Policy Manager is shared across all rules. If you define a Tcl variable in one rule, it is accessible in another rule also. We recommend that you use a unique prefix for local variables in each rule, to avoid polluting variables from different rules.*

## Editing advanced access policy rules

You write an advanced rule in one of the four situations described in *Understanding advanced access policy rule situations*, on page 11-16. These situations are:

- On the Advanced tab in the Expression popup screen of an action.
- On the Advanced tab in the Expression popup screen of the resource assign action.
- Using the custom variable and custom expression options in the variable assign action.
- Assigning a configuration variable to a custom expression in the variable assign action.

### To write an advanced access policy rule in an action

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. Add or edit an action.  
The action popup screen opens.
4. Click the Branch Rules tab.
5. Next to the Expression, click **change**.  
The rule editor popup screen opens.
6. Click the Advanced tab.

7. In the **Advanced** box, type the expression.
8. When you are finished, click **Finished**.
9. Click **Save**.

In this scenario, if the value returned by the expression is not zero, the rule is evaluated as true, and the access policy continues and follows the corresponding rule branch. If the value returned by the expression is zero, the rule is evaluated as false, and the access policy follows the branch assigned to the negative response (typically a fallback branch).

### **To write an advanced access policy rule in the resource assign action**

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. Add or edit a resource assign action.  
The resource assign popup screen opens.
4. Click the **Add New Entry** button.
5. In the Expression column, click **change**.  
The rule editor popup screen opens.
6. Click the Advanced tab.
7. In the **Advanced** box, type the expression.
8. When you are finished, click **Finished**.
9. Click **Save**.

In this scenario, the expression returns a value. If the return value is not zero, the resource assignment rule is true, and the access policy assigns the corresponding resource or ACL to the user. If the return value is zero, the resource assignment rule is evaluated as false, and the access policy does not assign the resource or ACL.

### To create a custom variable with an advanced access policy rule

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. Add or edit a variable assign action.  
The variable assign action popup screen opens.
4. Click the Branch Rules tab.
5. Next to the Expression, click **change**.  
The rule editor popup screen opens.
6. Under Assignment, click **change**.  
The Variable Assign popup screen opens.
7. In the **Custom Variable** box, type the new custom variable.
8. In the **Custom Expression** box, type the expression.
9. When you are finished, click **Finished**.
10. Click **Save**.

In this scenario, the custom expression returns a value that the variable assign action then assigns to the custom variable.

### To replace a configuration variable with a custom expression

1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. Add or edit a variable assign action.  
The variable assign action popup screen opens.
4. Click the Branch Rules tab.
5. Next to the Expression, click **change**.  
The rule editor popup screen opens.
6. Under Assignment, click **change**.  
The Variable Assign popup screen opens.
7. On the left side, select **Configuration Variable**.



8. From the **Name** list, select the name of the network access resource in which you want to overwrite the variable.
9. From the **Property** list, select the network access resource property you want to overwrite with a custom expression.
10. In the **Custom Expression** box, type the expression.
11. When you are finished, click **Finished**.
12. Click **Save**.

In this scenario, the expression returns a value that overwrites the value of the selected property from the network access resource.

## Example: Using a certificate field for logon name

In this example, the access policy parses the **CommonName (CN)** field from the client's SSL certificate, and the access policy uses part of that CN as the logon name. The result of this example, if the name field for the certificate includes **CN=Smith, OU=SBU,O=CompanyName,L=SanJose, ST=CA,C=US**, is that the data **Smith** is extracted from the name field, and the access policy passes this on as the logon name. Successive actions on this branch of the access policy can then use this logon name.

You can use the variable assignment agent to assign the value from the certificate's **CN** field to the value for the session variable **session.logon.last.username**, using the variable assignment agent.

## Writing the example code

The Tcl code for this example follows.

```
set cn_fields [split [mcget {session.ssl.cert.cn}] ","] ;

foreach field $cn_fields {
    if ($field contains "CN=") {
        set name [string range $field [expr { [string first "=" $field ] + 1 } ] end ] ;
        return $name ;
    }
} ;
```

*Figure 11.3 Tcl code to extract the logon name from a certificate field*

## Using this example

You assign the result of this example code to a custom variable called **session.logon.last.username** using the variable assign action.

### Add and edit the variable assign action

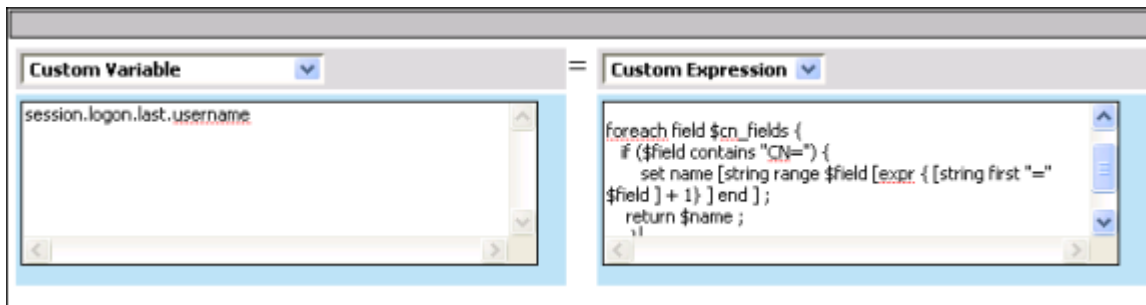
1. On the Main tab of the navigation pane, expand **Access Policy**, then click **Access Profiles**.  
The Access Profiles List screen opens.
2. In the profile list, find the access policy you want to edit, then click **Edit** in the Access Policy column.  
The visual policy editor opens in a new window or new tab, depending on your browser settings.
3. To add the variable assign action, click the plus sign ( **+** ) on an access policy branch.  
The Add Item popup screen opens.
4. On the Assignment tab, select **Variable Assign** and click **Add Item**.  
The Variable Assign action popup screen opens.

5. Click the **Add New Entry** button.
6. Under Assignment, next to **empty**, click **change**.  
The variable assignment editor popup screen opens.
7. In the **Custom Variable** box, type **session.logon.last.username**.
8. In the **Custom Expression** box, type the complete expression:

```
set cn_fields [split [mcget {session.ssl.cert.cn}] "," ] ;

foreach field $cn_fields {
    if ($field contains "CN=") {
        set name [string range $field [expr { [string first "=" $field ] + 1 } ] end ] ;
        return $name ;
    }
} ;
```

9. When you are finished, click **Finished**.
10. Click **Save**.



*Figure 11.4 Case study rule for Certificate CN in variable assign popup screen*





# 12

---

## Logging and Reporting

---

- Understanding logging
- Understanding log types
- Setting log levels
- Understanding reports
- Monitoring system and user information



## Understanding logging

Viewing and maintaining log messages is an important part of maintaining the Access Policy Manager®. Log messages inform you on a regular basis of the events that are happening on the system. Some of these events pertain to general events happening within the system, while other events are specific to the Access Policy Manager, such as stopping and starting Access Policy Manager system services.

The Access Policy Manager uses **syslog-ng** to log events. The **syslog-ng** utility is an enhanced version of the standard logging utility **syslog**.

The type of events messages available on the Access Policy Manager are:

- **Access Policy events**  
Access Policy event messages include logs pertinent to access policy, sso, network access, and portal access. To view access policy events, run Access Policy reports; expand **Access Policy** and click **Reports**.
- **Audit Logging**  
Audit event messages are those that the Access Policy Manager system logs as a result of changes made to its configuration.

For more information on other log events, refer to **BIG-IP® TMOS®: Concepts** guide available on the AskF5™ web site at <http://support.f5.com>.

## Introducing logging features

The logging mechanism on an Access Policy Manager® system includes several features designed to keep you informed of system events in the most effective way possible.

One of the primary features of logging is its ability to log different types of events, ranging from system events to access control events. Through the Access Policy Manager system auditing feature, you can even track and report changes that administrator makes to the BIG-IP® system configuration, such as adding a virtual server or changing an access policy. For more information, see *Understanding log content*, on page 12-2, and *Understanding log types*, on page 12-5.

When setting up logging on the Access Policy Manager, you can customize the logs by designating the minimum severity level, or log level, that you want the system to report when a type of event occurs. The **minimum log level** indicates the minimum severity level at which the system logs that type of event.

For examples of log levels, refer to *Setting log levels*, on page 12-7.

---

### ◆ Tip

*You can also configure the system to send email or to activate pager notification based on the priority of the logged event.*

## Understanding log content

The logs that the BIG-IP® system generates include several types of information. For example, some logs show a timestamp, host name, and service for each event. Moreover, logs sometimes include a status code, while the audit log shows a user name and a transaction ID corresponding to each configuration change. All logs contain an up to 2-line description of each event.

Table 12.1, following, displays the categories of information contained in the logs, and the specific logs in which the information is displayed.

Information Type	Explanation	Log Type
Timestamp	The time and date that the system logged the event message.	System Access Policy Audit
Log Level	Provides log level detail for each message.	Access Policy
Host	The host name of the system that logged the event message. Because this is typically the host name of the local machine, the appearance of a remote host name could be of interest.	System
Service	The service that generated the event.	System
Status code	The status code associated with the event. Note that only events logged by BIG-IP system components, and not operating system services, have status codes.	Access Policy
Session ID	The ID associated with the user session.	Access Policy
Description	The description of the event that caused the system to log the message.	System
User Name	The name of the user who made the configuration change.	Audit
Transaction	The identification number of the configuration change.	Audit
Event	Provides the description of the event so that it can be applicable to both Audit and Access policy logging.	Audit Access Policy

**Table 12.1** Log information categories and their descriptions

### ◆ Note

*For standalone clients, once a user has logged out and then logged back in, the sessions ID displays as invalid and remains as such in the **Notice** logs. The user is then assigned a new session ID. This is expected behavior of the system.*



## Modifying settings for the log database

By default, Access Policy Manager® writes logs to a database and to the /var/log/apm file. Access Policy Manager reports run against the data in the database. You can specify how frequently to remove the oldest logs from the database, control the maximum number of log entries that the database can hold, and remove all existing log records.

When log database tables are rotated, the oldest database table is dropped.

### To control database log rotation and maximum log entries

1. From the **Main** tab, select **Access Policy > Reports > Preferences**. The Preferences window opens. (If the **Enabled** check box is cleared for the **Write To Local Database** setting, the remaining settings are not available.)
2. In the **Log Rotation Period** box, type a number between 0 and 90. The default value is 0.  
When set to 0, log database tables are rotated only when the database contains the maximum number of log entries.  
When set to a value between 1 and 90, log database tables are rotated every *n* number of days. (If the maximum number of log entries is reached despite regular rotation, log database tables are rotated regardless.)
3. In the **Maximum Number Of Log Entries** box, type a number between 100000 and 5000000 (100,000 and 5,000,000). Do not type commas. The default value is 5000000.
4. Click **Update**.

### To remove all log data from the database

1. From the **Main** tab, select **Access Policy > Reports > Preferences**. The Preferences window opens.
2. Next to **Log Database Maintenance**, click **Delete**.  
All records are deleted from the reporting log database.

## Modifying settings for the log file

In addition to logging to a database, Access Policy Manager® logs to the /var/log/apm file. You might need the log file to help you troubleshoot a problem. If you configured logging to a remote server, you need Access Policy Manager to write to the log file for remote logging to work.

If you do not need or directly use the log file (for example, by searching them), you can stop Access Policy Manager from writing it.

### To configure APM to log to the database only

1. From the **Main** tab, select **Access Policy > Reports > Preferences**.

The Preferences window opens. (If the **Enabled** check box is cleared for the **Write To Local Database** setting, logs are not written to the database.)

2. Clear the **Enabled** box for **Write To APM Log File**.
3. Click **Update**.

---

◆ **Note**

*When Write To APM Log File is enabled, by default the file is rotated daily regardless of size.*

When running performance tests or under a very high traffic load, the /var/log/apm file can grow very large. While testing and otherwise, when a very high traffic load persists, you can mitigate the effect by disabling logging to /var/log/apm/ or by setting the log level to emergency only.

### **To disable logging to file for performance test or high traffic load**

Type this command:

```
tmsh modify sys db log.access.syslog value disable
```

### **To set log level to emergency for performance on the test or high traffic load**

Type this command:

```
tmsh modify sys db log.accesscontrol.level value emergency
```

### **To configure log rotation for the BIG-IP system**

To configure log rotation for the BIG-IP® system, use the **tmsh sys log-rotate** command. For more information about tmsh, refer to the *Traffic Management Shell (tmsh) Reference Guide*. You can also use the man pages for tmsh.

For more information about managing log files on the BIG-IP system, refer to <http://support.f5.com>.

## **About configuring logging to a remote server**

To configure remote logging, use the **tmsh modify /sys syslog remote-servers** command. For more information about the command, refer to the *Traffic Management Shell (tmsh) Reference Guide* on <http://support.f5.com>. You can also refer to the man pages for tmsh.

---

◆ **Note**

*The default syslog levels defined for the BIG-IP® system logs apply to local logs only; all syslog messages are sent to remote syslog servers.*

For information about filtering syslog messages sent to remote syslog servers, refer to <http://support.f5.com>.

## Understanding log types

The Access Policy Manager® can log two main event types:

- **Access policy:** Includes messages created during access policy validation, sso, network access, and portal access.
- **Audit:** Includes configuration changes.

You can view log information through the user interface or in log files.

- **Access policy events:** Provided that messages are logged in a local database (as they are by default), you can view them using Access Policy Manager reports. Also by default, messages are logged to the /var/log/apm file.
- **Audit events:** Messages are logged in the /var/log/audit file when audit logging is enabled.

## Logging system events

Many events that occur on Access Policy Manager® are operating system-related events, and do not specifically apply to the Access Policy Manager. The Access Policy Manager logs the messages for these events in the /var/log/messages file.

Using the Configuration utility, you can display these system messages. On the navigation pane, expand **System**, click **Logs**, and choose **System**.

## Auditing configuration changes

Audit logging is an optional feature that logs messages whenever there are changes made by the system. Such changes include the following items:

- User action
- System action
- Loading configuration data

The Access Policy Manager logs the messages for these auditing events in the /var/log/audit file.

Using the Configuration utility, you can display audit log messages. Table 12.2 shows some sample audit log entries. In this example, the first entry shows that user Janet enabled the audit logging feature, while the second and third entries show that user Matt designated the BIG-IP system to be a redundant system with a unit ID of **1**.

Timestamp	User Name	Transaction	Event
Mon Feb 14 03:34:45 PST 2008	janet	79255-1	DB_VARIABLE modified: name="config.auditing"
Mon Feb 14 03:35:06 PST 2008	matt	79609-1	DB_VARIABLE modified: name="failover.isredundant" value="true"
Mon Feb 14 03:35:06 PST 2008	matt	79617-1	DB_VARIABLE modified: name="failover.unitid" value="1"

**Table 12.2** *Sample audit log entries*

By default, audit logging is disabled. For information on enabling this feature, see *Setting log levels*, on page 12-7.

## Setting log levels

Using the Configuration utility, you can set log levels on auditing events and other types of events. The **minimum log level** indicates the minimum severity level at which the system logs that type of event. For more information, see *To set a minimum log level for local traffic events*, following.

For auditing events, you can set a log level that indicates the type of event that the system logs, such as the user-initiated loading of the Access Policy Manager system configurations, or system-initiated configuration changes. For more information, see *Setting log levels for auditing events*, on page 12-8.

### To set the log level for Access policy events

1. On the navigation pane, expand **System**, click **Logs**.  
The Logs screen opens.
2. On the menu bar, click **Configuration**, and select **Options**.  
The Logs screen changes to display the various logging options.
3. Depending on the type of log messages you want to control, scroll down to **Access Policy Logging**.
4. Select the log level for the selected component, and click **Update**.

The log levels that you can set on certain types of events, are sequenced from highest severity to lowest severity, like this:

- Emergency
- Alert
- Critical
- Error
- Warning
- Notice
- Informational
- Debug

### To set a minimum log level for local traffic events

1. On the navigation pane, expand **System**, and click **Logs**.  
The Logs screen opens.
2. On the menu bar, click **Configuration**, and select **Options**.  
The Logs screen changes to display the various logging options.
3. Scroll down to **Local Traffic Logging**.
4. Select the log level for the selected component, and click **Update**.

## Setting log levels for auditing events

An optional type of logging that you can enable is audit logging. Audit logging logs messages that pertain to configuration changes that users or services make to the BIG-IP® system configuration. This type of audit logging is known as **MCP audit logging**. (For more information, see *Auditing configuration changes*, on page 12-5.) Optionally, you can set up audit logging for any **tmsh** commands that users type on the command line.

For both MCP and **tmsh** audit logging, you can choose one of four log levels. In this case, the log levels do not affect the severity of the log messages; instead, they affect the initiator of the audit event.

For detailed information about auditing events, refer to the **BIG-IP® TMOS®: Concepts** on the AskF5™ web site at <http://support.f5.com>.

The log levels for MCP and **tmsh** audit logging are:

- **Disable**  
This turns audit logging off. This is the default value.
- **Enable**  
This causes the system to log messages for user-initiated configuration changes only.
- **Verbose**  
This causes the system to log messages for user-initiated configuration changes and any loading of configuration data.
- **Debug**  
This causes the system to log messages for all user-initiated and system-initiated configuration changes.

### To set a minimum log level for audit events

1. On the navigation pane, expand **System**, and click **Logs**.  
This Logs screen opens.
2. On the menu bar, click **Configuration** and select **Options**.
3. In the Audit Logging area near the bottom of the screen, select a log level from the **Audit Logging** list, which includes **MCP** and **tmsh**.
4. Click **Update**.

## Understanding reports

Access Policy Manager® supplies built-in reports and enables you to create custom reports. Built-in Session Reports enable you to review information about the sessions created on the system. With Access Policy Manager, the All Sessions report is the default report and displays first. You can set a different report as the default. After displaying the default report, you can then choose to run and view other built-in reports, such as Current Sessions. You can also define and run custom reports.

---

### ◆ Note

*For information about other built-in report types and report names, see online help.*

## Setting the default report

The default report runs each time you open the Reports window. If you do not set the default report, the All Sessions report functions as the default.

### To set the default report

1. Open the existing default report.
2. Run the report that you would like to use as the default.  
The report opens in a new tab.
3. In the report tab, click **Set to default report**.

## Displaying the All Sessions report

1. Select **Access Policy > Reports > View Reports**.  
The Report Parameters window opens, with a one-line description of the current default report and default Restrict by Time settings.
2. To display data for a non-default time period, select the appropriate **Restrict Time** settings.
3. Click **Run Reports**.  
The default report opens in the right pane.
4. If the All Sessions report is not displayed, perform these steps:
  - a) Scroll to the **Session Reports** list and select **All Sessions > Run**.  
The Report Parameters window opens, with a one-line description of the report and default Restrict by Time settings.
  - b) To display data for a non-default time period, select the appropriate **Restrict by Time** settings.
  - c) Click **Run Reports**.  
The All Sessions report is displayed in a new tab.

5. To view detailed information for a session, click a **Session ID**.  
A Session Details tab opens.

## Displaying session variables for current sessions

You can view session variables for any current session.

### To display session variables for a current session

1. Select **Access Policy > Reports > View Reports**.  
The Report Parameters window opens, with a one-line description of the default report along with default Restrict by Time settings.
2. Click **Run Reports**.  
The default report is displayed.
3. In the Reports Browser pane, scroll to the **Session Reports** list and select **Current Sessions > Run**.  
The Report Parameters window opens, with a one-line description of the report and default Restrict by Time settings.
4. To display data for a non-default time period, select the appropriate **Restrict by Time** settings.
5. Click **Run Reports**.  
The Current Sessions report is displayed in a new tab.
6. To view variables for a particular session, click the **View Session Variables** link in the Active column.  
A Session Variables page for the specific session ID opens in a new tab.

## Configuring and Running Custom Reports

Custom reports enable you to define the desired data, any constraints that you want to place on the data, and the sort order to use in a report. You can save, edit, and delete custom report definitions. In addition to running custom reports, you can export the report data to files.

### To configure a custom report

1. Select **Access Policy > Reports > View Reports**.  
The Report Parameters window opens, with a one-line description of the default report along with default Restrict by Time settings.
2. Click **Run Reports**.  
The default report is displayed.
3. In the Reports Browser pane, click **Custom Reports**.  
The Custom Reports area opens.



4. At the bottom of the Custom Reports area, click the **Create** icon.  
A Design Custom Report window opens with 3 tabs: Report Fields, Report Constraints, and Sort Fields.
5. Type a unique name in the **Report Name** field.
6. On the Report Fields tab, under these folders: Users, Resources, Session, and Access Policy, select fields by clicking check boxes.
7. Optionally, create constraints on the Report Constraints tab and specify a sort order on the Sort Fields tab. For more information, see online help.
8. Click **Save**.  
The Design Custom Report window closes. The name of the newly created custom report is displayed under Report Names in the Custom Reports area.

### To run a custom report

1. Select **Access Policy > Reports > View Reports**.  
The Report Parameters window opens, with a one-line description of the default report along with default **Restrict by Time** settings.
2. Click **Run Reports**.  
The default report is displayed.
3. In the Reports Browser pane, click **Custom Reports**.  
The Custom Reports area opens.
4. Select the report to run and click **Run Report**.  
A Custom Report Parameters window opens, displaying a default time range and any previously configured constraints.
5. Change the values that are displayed or leave them as is; click **Run Report**.  
The report displays in a new tab. The report results are not updated until you run the report again.

### To work with custom reports

1. Select **Access Policy > Reports > View Reports**.  
The Report Parameters window opens, with a one-line description of the default report along with default Restrict by Time settings.
2. Click **Run Reports**. The default report is displayed.
3. In the Reports Browser pane, click **Custom Reports**.  
The Custom Reports area opens.
4. The Custom Reports area lists any custom reports and displays icons labeled as follows:
  - Favorites - Puts the selected report on a list of favorites.
  - Delete - Deletes the selected report.

- Edit - Opens the Design Custom Report window for the selected report.
- Run - Runs the selected report.
- Export - Opens a dialog where you can select reports to export.
- Import - Opens a window where you can browse files to import.

## Monitoring system and user information

You can monitor overall system performance and Access Policy Manager® session information. The BIG-IP® system provides a dashboard that displays system statistics graphically, showing gauges and graphs, and you can view the same statistics in a table view. You can also view user session information specific to Access Policy Manager.

You can display the BIG-IP® system main dashboard from the navigation pane. Expand **Overview**, and click **Dashboard** tab. For more information on how to monitor overall system performance for the BIG-IP® system, refer to *Getting Started Guide: BIG-IP® Systems*.

The dashboard also includes online help for information about how to interpret statistics on each of the panels that appear on the screens. Click the question mark (?) in the upper right corner of any window to display the online help.

## Viewing the Access Policy Manager dashboard

In addition to the BIG-IP® system main dashboard, you can use the Access Policy Manager® dashboard to view specific Access Policy Manager users' session-based statistics, as well as throughput data.


With the Access Policy Manager dashboard, you can view the following information in four distinct panels:

- Active and new sessions
- Network access open and new connections
- Portal access cache information
- Access control list transactions

To view the dashboard, on the navigation pane, expand **Access Policy**, and click **Dashboard**.

---

### ◆ Tip

*By clicking the grid icon  in the upper left corner of each window, you can display the same information in a table format.*

## Monitoring active and new sessions

The top left panel of the Access Policy Manager dashboard displays the total and established connections for all current active and new sessions. This panel is called **Access Sessions**.

There are two tabs available for this panel:

- **Active Sessions:** Displays the number of active sessions.
- **New Sessions:** Displays the number of new sessions

You can view them in either real-time, or historical time ranges. You may want to view active sessions at various times of the day to determine the peak and select the best time to perform system maintenance, for example. If you notice that the total number of sessions peaked while the total number of established sessions remain low, this may be an indication that a possible malicious attack is occurring in your network environment.

## Monitoring portal access cache information

The bottom left panel of the Access Policy Dashboard displays cache effectiveness by comparing the three available metrics. This panel is called **Portal Access**. There are currently no tabs available for this panel, but the metrics include:

- **Client Requests:** Displays the total cache requests from the client.
- **Request Served from RamCache:** Displays the total number of cache hits.
- **Requests Missed from RamCache:** Displays the total number of cache misses.

Hits and misses are derived by subtracting the server responses from the client responses. A server response indicates that the requested information was not in cache.

## Monitoring network access throughput and connections

The right top panel of the Access Policy dashboard displays throughput data for the amount of traffic through the network access tunnels, as well as displays open and new connections. This panel is called **Network Access**.

You can view throughput numbers to and from the client, as well as the overall throughput for network access traffic.

Use this panel to determine how much traffic is going through the tunnels, and how many people are generating that traffic. For example, if there are two tunnels, and those particular users are generating gigabytes of traffic, you may want to further investigate the activities on those tunnels.

This panel is also useful as a good indicator for peaked traffic to determine the best time to perform system maintenance

There are four tabs available for this panel:

- **Throughput:** Displays the amount of throughput for data transfers through the network access tunnels.
- **Open Connections:** Displays the number of open connections through the network access tunnels.
- **New Connections:** Displays the number of new connections through the network access tunnels.
- **Compression:** Displays the compression level through the network access tunnel. The Compression tab provides a gauge as well as a chart.

## Monitoring access control list information

The bottom right panel of the Access Policy Dashboard displays ACL activities.

There is one tab available for this panel:

**ACL Actions:** Displays the action that the access control list takes when an access control entry is encountered.





# 13

---

## Configuring SNMP

---

- Introducing SNMP administration
- Configuring the SNMP agent
- Working with SNMP MIB files
- Collecting performance data





## Introducing SNMP administration

*Simple Network Management Protocol (SNMP)* is an industry-standard protocol that gives a standard SNMP management system the ability to remotely manage a device on the network. One of the devices that an SNMP management system can manage is a Access Policy Manager system. The SNMP versions that the Access Policy Manager system supports are: SNMP v1, SNMP v2c, and SNMP v3. The Access Policy Manager system implementation of SNMP is based on a well-known SNMP package, Net-SNMP, which was formerly known as UCD-SNMP.

## Reviewing an industry-standard SNMP implementation

A standard SNMP implementation consists of an *SNMP manager*, which runs on a management system and makes requests to a device, and an *SNMP agent*, which runs on the managed device and fulfills those requests. SNMP device management is based on the standard management information base (MIB) known as MIB-II, as well as object IDs and MIB files.

- The *MIB* defines the standard objects that you can manage for a device, presenting those objects in a hierarchical, tree structure.
- Each object defined in the MIB has a unique object ID (OID), written as a series of integers. An *OID* indicates the location of the object within the MIB tree.
- A set of MIB files resides on both the SNMP manager system and the managed device. *MIB files* specify values for the data objects defined in the MIB. This set of MIB files consists of standard SNMP MIB files and enterprise MIB files. *Enterprise* MIB files are those MIB files that pertain to a particular company, such as F5 Networks, Inc.

Typical SNMP tasks that an SNMP manager performs include polling for data about a device, receiving notifications from a device about specific events, and modifying writable object data.

## Reviewing the Access Policy Manager system SNMP implementation

To comply with the standard SNMP implementation, the Access Policy Manager system includes both an SNMP agent, a set of standard SNMP MIB files, and a set of enterprise MIB files (those that are specific to the Access Policy Manager system). The enterprise MIB files typically reside on both the Access Policy Manager system, and on the system running the SNMP manager. Fortunately, you can use the browser-based Configuration utility to download the enterprise MIB files to your SNMP manager.

Using the Access Policy Manager system implementation of SNMP, the SNMP manager can perform these distinct functions:

- Poll for information (such as performance metrics).
- Receive notification of specific events that occur on the Access Policy Manager system.
- Set data for SNMP objects that have a read/write access type.

The last item in the list refers to the ability of an SNMP manager system to enable or disable various Access Policy Manager system objects such as virtual servers and nodes. Specifically, you can use SNMP to:

- Enable or disable a virtual server
- Enable or disable a virtual address
- Enable or disable a node
- Enable or disable a pool member
- Set a node to an **up** or **down** state
- Set a pool member to an **up** or **down** state
- Reset statistical data for all Access Policy Manager objects

## Summarizing SNMP configuration on the Access Policy Manager system

Before an SNMP manager can manage a Access Policy Manager system remotely, you must perform a few configuration tasks on the Access Policy Manager system, using the Access Policy Manager system's Configuration utility. After you have performed these configuration tasks, you can use standard SNMP commands on the remote manager system to manage the Access Policy Manager system.

The configuration tasks you perform are:

### ◆ **Configuring the SNMP agent**

There are a number of things you can do to configure the SNMP agent on the Access Policy Manager system. For example, you can allow client access to information that the SNMP agent collects, and you can configure the way that the SNMP agent handles SNMP traps. **Traps** are definitions of unsolicited notification messages that the Access Policy Manager alert system and the SNMP agent send to the SNMP manager when certain events occur.

### ◆ **Downloading MIB files**

You can download two sets of MIB files to your remote manager system: the standard SNMP MIB files and the enterprise MIB files. From the navigation pane, expand **Overview**, and click **Welcome**. From the Welcome screen, scroll down to **Downloads**.

## Configuring the SNMP agent

To configure the SNMP agent on the Access Policy Manager system, you can use the Configuration utility. Configuring the SNMP agent means performing the following tasks:

- **Configuring Access Policy Manager system information**  
Specify a system contact name and the location of the Access Policy Manager system.
- **Configuring client access to the SNMP agent**  
Configure the Access Policy Manager system to allow access to the SNMP agent from an SNMP manager system.
- **Controlling access to SNMP data**  
Assign access levels to SNMP communities or users, to control access to SNMP data.
- **Configuring Traps**  
Enable or disable traps and specify the destination SNMP manager system for SNMP traps.

You can use the Configuration utility to configure the following information:

- **Contact Information**  
The contact information is a MIB-II simple string variable defined by almost all SNMP boxes. The contact name usually contains a user name, as well as an email address.
- **Machine Location**  
The machine location is a MIB-II variable that almost all machines support. It is a simple string that defines the location of the machine.

### To configure system information

1. On the Main tab of the navigation pane, expand **System**, and click **SNMP**.  
The SNMP Agent Configuration screen opens.
2. In the Global Setup area, fill in the boxes.  
For more information, see the online help.
3. Click **Update**.

## Configuring client access

An SNMP *client* refers to any system running the SNMP manager software for the purpose of remotely managing the Access Policy Manager system. To set up client access to the Access Policy Manager system, you specify the IP or network addresses (with netmask as required) from which the SNMP agent can accept requests. (By default, SNMP is enabled only for the Access Policy Manager system loopback interface, **127.0.0.1**.)

### To allow client access to the SNMP agent

1. On the Main tab of the navigation pane, expand **System**, and click **SNMP**.  
The SNMP Agent Configuration screen opens.
2. For the **Client Allow List** setting, select **Host** or **Network**, depending on whether the IP address you specify is a host system or a subnet.
3. Type the following information:
  - In the **Address** box, type an IP address or network address from which the SNMP agent can accept requests.
  - If you selected **Network** in step 2, type the netmask in the **Mask** box.
4. Click the **Add** button to add the host or network address to the list of allowed clients.
5. Click **Update**.

## Controlling access to SNMP data

To better control access to SNMP data, you can assign an access level to an SNMP v1 or v2c community, or to an SNMP v3 user.

There is a default access level for communities, and this access level is read-only. This means that you cannot write to an individual data object that has a read/write access type until you change the default read-only access level of the community or user.

The way to modify this default access level is by using the Configuration utility to grant read/write access to either a community (for SNMP v1 and v2c) or a user (SNMP v3), for a given OID.

When you set the access level of a community or user to read/write, and an individual data object has a read-only access type, access to the object remains read-only. In short, the access level or type that is the most secure takes precedence when there is a conflict. Table 13.1 illustrates this point.

If the access type of an object is...	And you set the access level of a community or user to...	Then access to the object is...
Read-only	Read-only	Read-only
	Read/write	Read-only
Read/write	Read-only	Read-only
	Read/write	Read/write

*Table 13.1 Access control for SNMP data*

### To grant community access to SNMP data (v1 or v2c only)

1. On the Main tab of the navigation pane, expand **System**, and click **SNMP**.  
The SNMP Agent Configuration screen opens.
2. From Agent menu, choose Access (v1, v2c).  
The SNMP Access screen opens.
3. In the upper-right corner of the screen, click **Create**.  
The New Access Record screen opens.
4. Select the type of address to which the access record applies.
5. In the **Community** box, type the name of the SNMP community for which you are assigning an access level (in step 8).
6. In the **Source** box, type the source IP address.
7. In the **OID** box, type the OID for the top-most node of the SNMP tree to which the access applies.

8. For the **Access** setting, select an access level, either **Read Only** or **Read/Write**. (This access level applies to the community name you specified in step 6.)
9. Click **Finished**.

### To grant access to SNMP data (v3 only)

1. On the Main tab of the navigation pane, expand **System**, and click **SNMP**.  
The SNMP Agent Configuration screen opens.
2. From Agent menu, choose Access (v3).  
The SNMP Access screen opens.
3. In the upper-right corner of the screen, click **Create**.  
The New Access Record screen opens.
4. In the **User Name** box, type a user name for which you are assigning an access level (in step 8).
5. For the **Authentication** setting, select a type of authentication to use, and then type and confirm the user's password.
6. For the **Privacy** setting, select a privacy protocol, and then do either of the following:
  - Type and confirm the user's password.
  - Check the **Use Authentication Password** box.
7. In the **OID** box, type the object identifier (OID) for the top-most node of the SNMP tree to which the access applies.
8. For the **Access** setting, select an access level, either **Read Only** or **Read/Write**. (This access level applies to the user name that you specified in step 5).
9. Click **Finished**.

---

#### **WARNING**

*You must remember to configure both **authentication** and **privacy** settings to use SNMPv3. Otherwise, an error occurs and SNMPv3 will not work properly.*

---

#### **Note**

*SNMPv3 currently supports **AuthPriv** setting only. It does not support **AuthNoPrivacy**.*

When you use the Configuration utility to assign an access level to a community or user, the utility updates the **snmpd.conf** file, assigning only a single access setting to the community or user. There might be times,

however, when you want to configure more sophisticated access control. To do this, you must edit the `/config/snmp/snmpd.conf` file directly, instead of using the Configuration utility.

For example, Figure 13.1 shows a sample `snmpd.conf` file when you use the Configuration utility to grant read/write access to a community.

```
rocommunity public default
rwcommunity public1 127.0.0.1 .1.3.6.1.4.1.3375.2.2.10.1
```

**Figure 13.1** Sample access-control assignments in the `snmpd.conf` file

In this example, the string `rocommunity` identifies a community named **public** as having the default read only access level (indicated by the strings **ro** and **default**). This read only access level prevents any allowed SNMP manager in community **public** from modifying a data object, even if the object has an access type of read/write.

The string `rwcommunity` identifies a community named **public1** as having a read/write access level (indicated by the string **rw**). This read/write access level allows any allowed SNMP manager in community **public1** to modify a data object under the tree node **1.2.6.1.4.1.3375.2.2.10.1** (**ItmVirtualServ**) on the local host **127.0.0.1**, if that data object has an access type of read/write.

For more information, see the man page for the `snmpd.conf` file.

## Configuring traps

On the Access Policy Manager system, **traps** are definitions of unsolicited notification messages that the Access Policy Manager alert system and the SNMP agent send to the SNMP manager when certain events occur on the Access Policy Manager system. Configuring SNMP traps on a Access Policy Manager system means configuring the way that the Access Policy Manager system handles traps, as well as setting the destination for notifications that the alert system and the SNMP agent send to an SNMP manager.

The Access Policy Manager system stores traps in two specific files:

- **/etc/alertd/alert.conf**  
Contains default SNMP traps.
- **/config/user\_alert.conf**  
Contains user-defined SNMP traps.

### ◆ Important

*Do not add or remove traps from the `/etc/alertd/alert.conf` file.*

You use the Configuration utility to configure traps, that is, enable traps and set trap destinations. When you configure traps, the Access Policy Manager system automatically updates the `alert.conf` and `user_alert.conf` files.

## Enabling traps for specific events

You can configure the SNMP agent on the Access Policy Manager system to send, or refrain from sending, notifications when the following events occur:

- The SNMP agent on the Access Policy Manager system stops or starts. By default, this trap is enabled.
- The Access Policy Manager system receives an authentication warning, generated when a client system attempts to access the SNMP agent. By default, this trap is disabled.
- The Access Policy Manager system receives any type of warning. By default, this trap is enabled.

### To enable traps for specific events

1. On the Main tab of the navigation pane, expand **System**, and click **SNMP**.  
This opens the SNMP Agent Configuration screen.
2. From the Traps menu, choose Configuration.  
This displays the SNMP Trap Configuration screen.
3. To send traps when someone starts or stops the SNMP agent, verify that the **Agent Start/Stop** box is checked.
4. To send notifications when authentication warnings occur, check the **Agent Authentication** box.
5. To send notifications when certain warnings occur, verify that the **Device** box is checked.
6. Click **Update**.

## Setting the trap destination

In addition to enabling certain traps for certain events, you must specify the destination SNMP manager to which the Access Policy Manager system should send notifications. For SNMP versions 1 and 2c only, you specify a destination system by providing the community name to which the Access Policy Manager system belongs, the IP address of the SNMP manager, and the target port number of the SNMP manager.

### ◆ Important

---

*If you are using SNMP V3 and want to configure a trap destination, you do not use the SNMP screens within the Configuration utility. Instead, you configure the `snmpd.conf` file. For more information, see the man page for the `snmpd.conf` file.*

### To specify a trap destination

1. On the Main tab of the navigation pane, expand **System**, and click **SNMP**.  
The SNMP Agent Configuration screen opens.



2. From the Traps menu, choose Destination.  
The SNMP Destination screen opens.
3. In the upper-right corner, click **Create**.  
The New Trap Record screen opens.
4. For the **Version** setting, select an SNMP version number.
5. In the **Community** box, type the community name for the SNMP agent running on the Access Policy Manager system.
6. In the **Destination** box, type the IP address of the SNMP management system.
7. In the **Port** box, type the SNMP management system port number that is to receive the traps.
8. Click **Finished**.

## Working with SNMP MIB files

As described earlier, *MIB files* define the SNMP data objects contained in the SNMP MIB. There are two sets of MIB files that typically reside on the Access Policy Manager system and the SNMP manager system: enterprise MIB files (that is, F5-specific MIB files) and standard SNMP MIB files.

Both sets of MIB files are already present on the Access Policy Manager system, in the directory `/usr/share/snmp/mibs`. However, you still need to download them to your SNMP manager system. You can download these MIB files from the Welcome screen of the browser-based Configuration utility. For more information, see *Downloading SNMP MIB files*.

To make MIB-II as clear as possible, we have implemented the SNMP feature so that you use MIB-II for gathering standard Linux data only. You cannot use MIB-II to gather data that is specific to the Access Policy Manager system and instead must use the F5 enterprise MIB files. All OIDs for Access Policy Manager system data are contained in the F5 enterprise MIB files, including all interface statistics (**1.3.6.1.4.1.3375.2.1.2.4** (`sysNetwork.sysInterfaces`)).

---

◆ **Note**

*All Access Policy Manager system statistics are defined by 64-bit counters. Thus, because only SNMP v2c supports 64-bit counters, your management system needs to use SNMP v2c to query Access Policy Manager system statistics data.*

---

## Downloading SNMP MIB files

The enterprise MIB files that you can download to the SNMP manager system are:

- **F5-BIGIP-COMMON-MIB.txt**  
This MIB file contains common information and all notifications (traps).
- **F5-BIGIP-LOCAL-MIB.txt**  
This is an enterprise MIB file that contains specific information for properties associated with specific Access Policy Manager system features related to local traffic manager (such as virtual servers, pools, and SNATs).
- **F5-BIGIP-SYSTEM-MIB.txt**  
The **F5-BIGIP-SYSTEM-MIB.txt** MIB file includes global information on system-specific objects.
- **F5-BIGIP-APM-MIB.txt**  
This MIB file contains specific information for properties associated with viewing and accessing access profile and secure connectivity statistics.

To view the set of standard SNMP MIB files that you can download to the SNMP manager system, list the contents of the Access Policy Manager system directory `/usr/share/snmp/mibs`.

### To download MIB files

1. On the navigation pane, select the **About** tab.
2. Scroll to the Downloads section, and locate the **SNMP MIBs** section.
3. Click the appropriate link for F5 MIB or Net-SNMP MIB files.
4. Follow the instructions on the screen to complete the download.

## Understanding the enterprise MIB files

Once you have downloaded all of the necessary MIB files, you should familiarize yourself with the contents of the enterprise MIBs, for purposes of managing the Access Policy Manager system and troubleshooting Access Policy Manager system events.

---

### ◆ Note

*To manage a Access Policy Manager system with SNMP, you need to use the standard set of SNMP commands. For information on SNMP commands, consult your favorite third-party SNMP documentation, or visit the web site <http://net-snmp.sourceforge.net>.*

The Access Policy Manager system includes a set of enterprise MIB files:

- **F5-BIGIP-COMMON-MIB.txt**
- **F5-BIGIP-LOCAL-MIB.txt**
- **F5-BIGIP-APM-MIB.txt**
- **F5-BIGIP-SYSTEM-MIB.txt**

These MIB files contain information that you can use for your remote management station to poll the SNMP agent for Access Policy Manager system-specific information, receive Access Policy Manager system-specific notifications, or set Access Policy Manager system data.

## Using the F5-BIGIP-COMMON-MIB.txt file

The **F5-BIGIP-COMMON-MIB.txt** file is an enterprise MIB file that contains objects pertaining to any common information, as well as the F5-specific SNMP traps.

All F5-specific traps are contained within this MIB file. You can identify the traps within this MIB file by viewing the file and finding object names that show the designation NOTIFICATION-TYPE.

When an F5-specific trap sends a notification to the SNMP manager system, the SNMP manager system receives a text message describing the event or problem that has occurred.

To see all available MIB objects in this MIB file, you can view the **F5-BIGIP-COMMON-MIB.txt** file in the directory **/usr/share/snmp/mibs** on the Access Policy Manager system.

## Using the F5-BIGIP-LOCAL-MIB.txt file

The **F5-BIGIP-LOCAL-MIB.txt** file is an enterprise MIB file that contains information that an SNMP manager system can access for the purpose of managing local application traffic. For example, you can:

- View the maximum number of entries that a node can have open at any given time.
- Get a pool name.
- View the current active members for a load balancing pool.
- Reset pool statistics
- Get profile information such as the total number of concurrent authentication sessions.

In general, you can use this MIB file to get information on any local traffic manager object (virtual servers, pools, nodes, profiles, SNATs, health monitors, and iRules). You can also reset statistics for any of these objects.

To see all available enterprise MIB objects for local traffic manager, you can view the **F5-BIGIP-LOCAL-MIB.txt** file in the directory **/usr/share/snmp/mibs** on the Access Policy Manager system.

## Using the F5-BIGIP-SYSTEM-MIB.txt file

The **F5-BIGIP-SYSTEM-MIB.txt** file is an enterprise MIB file that describes objects representing common system information. Examples of information in this MIB file are global statistic data, network information, and platform information. Some of the data in this MIB file is similar to that defined in MIB-II, but is not exactly the same.

Table 13.2 shows standard MIB-II objects and the F5-specific objects that approximately correspond to them.

MIB-II Category or Object	F5-BIGIP-SYSTEM-MIB Object Name
MIB-II	f5.bigipSystem
interfaces	sysNetwork.sysInterfaces.sysInterface sysNetwork.sysInterfaces.sysInterfaceStat sysNetwork.sysInterfaces.sysInterfaceMediaOptions
ip	sysGlobalStats.sysGlobalIpStat
ip.AddrTable	sysNetwork.sysSelfIp
ip.RouteTable	sysNetwork.sysRoute
ip.ipNetToMediaTable	sysNetwork.sysArpNdp
icmp	sysGlobalStats.sysGlobalIcmpStat
tcp	sysGlobalStats.sysGlobalTcpStat
udp	sysGlobalStats.sysGlobalUdpStat
transmission/dot3.dot3StatTable transmission/dot3.dot3CollTable	sysNetwork.sysTransmission.sysDot3Stat
dot1dBridge.dot1dBase	sysNetwork.sysDot1dBridge
dot1dBridge.dot1dStp	sysNetwork.sysSpanningTree.sysStpBridgeStat sysNetwork.sysSpanningTree.sysStpBridgeTreeStat sysNetwork.sysSpanningTree.sysInterfaceStat sysNetwork.sysSpanningTree.sysInterfaceTreeStat
dot1dBridge.dot1dTp	sysGlobalAttr.VlanFDBTimeout
dot1dBridge.dot1dTpFdbTable	sysNetwork.sysL2
dot1dTpPortTable	sysNetwork.sysInterfaces.sysInterfaceStat
dot1dStaticTable	Not supported.
ifMIB/ifMIBObjects.ifXTable	sysNetwork.sysInterfaces.sysIfxStat

**Table 13.2** F5-BIGIP-SYSTEM-MIB objects and their relationship to MIB-II objects

To see all available enterprise MIB system objects, you can view the **F5-BIGIP-SYSTEM-MIB.txt** file in the directory **/usr/share/snmp/mibs** on the Access Policy Manager system.

## Using the RMON-MIB.txt file

One of the MIB files that the Access Policy Manager system provides is the Remote network Monitoring (RMON) MIB file, **RMON-MIB.txt**. This file is the standard RMON MIB file. However, the implementation of RMON on the Access Policy Manager system differs slightly from the standard RMON implementation, in these ways:

- The Access Policy Manager system implementation of RMON supports four of the nine RMON groups. The four supported RMON groups are: statistics, history, alarms, and events.
- The **RMON-MIB.txt** file monitors the Access Policy Manager system interfaces (that is, **sysIfIndex**), and not the standard Linux interfaces.
- For hardware reasons, the packet-length-specific statistics in the RMON statistics group offer combined transmission and receiving statistics only. This behavior differs from the behavior described in the definitions of the corresponding object IDs.

To understand how RMON operates for a Access Policy Manager system, you can view the **RMON-MIB.txt** file in the directory **/usr/share/snmp/mibs** on the Access Policy Manager system.

## Using the F5-BIGIP-APM-MIB file

As mentioned earlier, this MIB file contains specific information associated with viewing and accessing access profile and secure connectivity statistics.

For a list of the type of objects used to view both access policy and secure connectivity statistics, refer to Chapter 11, *Logging and Reporting*.

## Collecting performance data

The Configuration utility on the Access Policy Manager system displays graphs showing performance metrics for the system. However, you can also use SNMP to collect the same information.

The types of performance metrics that you can gather using SNMP are:

- Megabytes of memory being used
- Number of active connections
- Number of new connections
- Throughput in bits per second
- Number of HTTP requests
- CPU use
- Number of current active sessions

Each type of metric has one or more SNMP object IDs (OIDs) associated with it. To gather performance data, you specify these OIDs with the appropriate SNMP command.

For example, the following SNMP command collects data on current memory use, where **public** is the community name and **bigip** is the host name of the Access Policy Manager system:

```
snmpget -c public bigip sysGlobalStat.sysStatMemoryUsed.0
```

For some types of metrics, such as memory use, simply issuing an SNMP command with an OID gives you the information you need. For other types of metrics, the data that you collect with SNMP is not useful until you perform a calculation on it.

For example, to determine the throughput rate of client bits coming into the Access Policy Manager system, you must perform the following calculation on the data that you collect with the OID shown:

```
( sysStatClientBytesIn (.1.3.6.1.4.1.3375.2.1.1.2.1.3)*8 ) / time
```

This calculation takes the data resulting from specifying the OID **sysStatClientBytesIn**, multiplies the value by **8**, and divides it by the elapsed time.

The following sections contain tables that list:

- The performance data that the Configuration utility displays
- The OIDs that you can use to collect the performance data
- The calculations that you must perform to interpret the performance data that you collect

### ◆ Note

*If an OID that is listed in any of the following sections does not show a calculation, then no calculation is required.*

## Collecting data on memory use

You can use an SNMP command with OIDs to gather data on the number of megabytes of memory currently being used on the Access Policy Manager system. Table 13.3 shows the OIDs that you need to specify to gather data on the current memory use. To collect memory use data, you do not need to perform a calculation on the collected data.

Performance Graph (Configuration utility)	Graph Metric	Required SNMP OID
Memory Used	TMM Mem Usage	sysStatMemoryUsed (.1.3.6.1.4.1.3375.2.1.1.2.1.45)
	Host Mem Usage	sysHostMemoryUsed (.1.3.6.1.4.1.3375.2.1.7.2)

**Table 13.3** Required OIDs for collecting metrics on memory use

## Collecting data on active connections

You can use SNMP commands with various OIDs to gather data on the number of active connections on the Access Policy Manager system. Table 13.4 shows the OIDs that you need to specify to gather data on active connections. In this case, you do not need to perform any calculations on the collected data.

Performance Graph (Configuration utility)	Graph Metrics	Required SNMP OIDs
Active Connections (summary graph)	Connections	sysStatClientCurConns (.1.3.6.1.4.1.3375.2.1.1.2.1.8)
	client	sysStatClientCurConns (.1.3.6.1.4.1.3375.2.1.1.2.1.8)
Active Connections (detailed graph)	server	sysStatServerCurConns (.1.3.6.1.4.1.3375.2.1.1.2.1.15)
	Client Bits Out	(sysStatClientBytesOut (.1.3.6.1.4.1.3375.2.1.1.2.1.5) *8) / time
	ssl client	sysClientsslStatCurConns (.1.3.6.1.4.1.3375.2.1.1.2.9.2)
	ssl server	sysServersslStatCurConns (.1.3.6.1.4.1.3375.2.1.1.2.10.2)

**Table 13.4** Required OIDs for collecting metrics on active connections



## Collecting data on new connections

You can use SNMP commands with various OIDs to gather data on the number of new connections on the Access Policy Manager system. Table 13.5 shows the OIDs that you need to specify to gather data on new connections, along with the calculations that you must perform on the collected data.

Performance Graph (Configuration utility)	Graph Metrics	Required SNMP OIDs and the required calculations
New Connections (summary graph)	Client Connections	sysStatClientTotConns (.1.3.6.1.4.1.3375.2.1.1.2.1.7)
	Client Accepts	sysTcpStatAccepts (.1.3.6.1.4.1.3375.2.1.1.2.12.6) / time
	Server Connects	sysTcpStatConnects (.1.3.6.1.4.1.3375.2.1.1.2.12.8) / time
Total New Connections (detailed graph)	Client Connections	sysStatClientTotConns (.1.3.6.1.4.1.3375.2.1.1.2.1.7) / time
	Server Connections	sysStatServerTotConns (.1.3.6.1.4.1.3375.2.1.1.2.1.14) / time
New PVA Connections (detailed graph)	pva client	sysStatPvaClientTotConns (.1.3.6.1.4.1.3375.2.1.1.2.1.21) / time
	pva server	sysStatPvaServerTotConns (.1.3.6.1.4.1.3375.2.1.1.2.1.28) / time
New SSL Connections (detailed graph)	SSL Client	( sysClientsslStatTotNativeConns (.1.3.6.1.4.1.3375.2.1.1.2.9.6) + sysClientsslStatTotCompatConns (.1.3.6.1.4.1.3375.2.1.1.2.9.9) ) / time
	SSL Server	( sysServersslStatTotNativeConns (.1.3.6.1.4.1.3375.2.1.1.2.10.6) + sysServersslStatTotCompatConns (.1.3.6.1.4.1.3375.2.1.1.2.10.9) ) / time
New Accepts/Connects (detailed graph)	Client Accepts	sysTcpStatAccepts (.1.3.6.1.4.1.3375.2.1.1.2.12.6) / time
	Server Connects	sysTcpStatConnects (.1.3.6.1.4.1.3375.2.1.1.2.12.8) / time

**Table 13.5** Required OIDs for collecting metrics on new connections

## Collecting data on throughput

You can use SNMP commands with various OIDs to gather data on the throughput rate on the Access Policy Manager system, in terms of bits per second. Table 13.6 shows the OIDs that you need to specify to gather data on throughput rate, along with the calculations that you must perform on the collected data.

Performance Graph (Configuration utility)	Graph Metrics	Required SNMP OIDs and the required calculations
Throughput (summary graph)	Client Bits	$(\text{sysStatClientBytesIn} (.1.3.6.1.4.1.3375.2.1.1.2.1.3) + \text{sysStatClientBytesOut} (.1.3.6.1.4.1.3375.2.1.1.2.1.5)) * 8 / \text{time}$
	Server Bits	$(\text{sysStatServerBytesIn} (.1.3.6.1.4.1.3375.2.1.1.2.1.10) + \text{sysStatServerBytesOut} (.1.3.6.1.4.1.3375.2.1.1.2.1.12)) * 8 / \text{time}$
Throughput (detailed graph)	Client Bits In	$(\text{sysStatClientBytesIn} (.1.3.6.1.4.1.3375.2.1.1.2.1.3)) * 8 / \text{time}$
	Client Bits Out	$(\text{sysStatClientBytesOut} (.1.3.6.1.4.1.3375.2.1.1.2.1.5)) * 8 / \text{time}$
	Server Bits In	$(\text{sysStatServerBytesIn} (.1.3.6.1.4.1.3375.2.1.1.2.1.10)) * 8 / \text{time}$
	Server Bits Out	$(\text{sysStatServerBytesOut} (.1.3.6.1.4.1.3375.2.1.1.2.1.12)) * 8 / \text{time}$

**Table 13.6** Required OIDs for collecting metrics on throughput

## Collecting data on HTTP requests

You can use SNMP commands with various OIDs to gather data on the number of current HTTP requests on the Access Policy Manager system, in terms of requests per second. Table 13.7 shows the OID that you need to specify to gather data on HTTP requests, along with the calculations that you must perform on the collected data.

Performance Graph (Configuration utility)	Graph Metric	Required SNMP OID and the required calculation
HTTP Requests	HTTP Requests	$\text{sysStatHttpRequests} (.1.3.6.1.4.1.3375.2.1.1.2.1.56) / \text{time}$

**Table 13.7** Required OIDs for collecting metrics on HTTP requests

## Collecting data on RAM Cache utilization

You can use an SNMP command with various OIDs to gather data on RAM cache utilization. Table 13.8 shows the OIDs that you need to specify to gather this data.

Performance Graph (Configuration utility)	Graph Metric	Required SNMP OID
RAM Cache Utilization	Hit Rate	$\text{sysHttpStatRamcacheHits} (.1.3.6.1.4.1.3375.2.1.1.2.4.46) / (\text{sysHttpStatRamcacheHits} (.1.3.6.1.4.1.3375.2.1.1.2.4.46) + \text{sysHttpStatRamcacheMisses} (.1.3.6.1.4.1.3375.2.1.1.2.4.47)) * 100$
	Byte Rate	$\text{sysHttpStatRamcacheHitBytes} (.1.3.6.1.4.1.3375.2.1.1.2.4.49) / (\text{sysHttpStatRamcacheHitBytes} (.1.3.6.1.4.1.3375.2.1.1.2.4.49) + \text{sysHttpStatRamcacheMissBytes} (.1.3.6.1.4.1.3375.2.1.1.2.4.50)) * 100$
	Eviction Rate	$\text{sysHttpStatRamcacheEvictions} (.1.3.6.1.4.1.3375.2.1.1.2.4.54) / (\text{sysHttpStatRamcacheHits} (.1.3.6.1.4.1.3375.2.1.1.2.4.46) + \text{sysHttpStatRamcacheMisses} (.1.3.6.1.4.1.3375.2.1.1.2.4.47)) * 100$

**Table 13.8** Required OIDs for collecting metrics on RAM Cache utilization

## Collecting data on CPU use

You can use SNMP commands with various OIDs to gather data on CPU use on the Access Policy Manager system. Specifically, you can gather data for two different graph metrics: TMM CPU Usage and CPU[0-n].

To gather the data for each of these metrics, you must perform some polling and calculations. First, for each metric type (for example, **sysStatTmTotalCycles**), you must perform two separate polls, at ten-second intervals. Then, you must calculate the delta of the two polls. Finally, you must use these delta values to perform the calculation shown in Table 13.9. The two sections following the table contain the specific procedures you use to calculate metrics for TMM CPU Usage and CPU[0-n] metric types.

Performance Graph (Configuration utility)	Graph Metric	Required SNMP OIDs and the required calculation
CPU Usage	CPU[0-n]	$(\text{DeltaCpuUser} + \text{DeltaCpuNice} + \text{DeltaCpuSystem}) / (\text{DeltaCpuUser} + \text{DeltaCpuNice} + \text{DeltaCpuIdle} + \text{DeltaCpuSystem} + \text{DeltaCpuIrq} + \text{DeltaCpuSoftirq} + \text{DeltaCpuIowait})$
	TMM CPU Usage	$(\text{DeltaTmTotalCycles} - (\text{DeltaTmIdleCycles} + \text{DeltaTmSleepCycles})) / \text{DeltaTmTotalCycles} * 100$

**Table 13.9** Required OIDs for collecting metrics on CPU use

**To calculate the CPU[0-n] metric**

1. Perform two separate polls of each of the following OIDs:

- **sysHostCpuUser** (.1.3.6.1.4.1.3375.2.1.7.2.2.1.3)
- **sysHostCpuNice** (.1.3.6.1.4.1.3375.2.1.7.2.2.1.4)
- **sysHostCpuSystem** (.1.3.6.1.4.1.3375.2.1.7.2.2.1.5)
- **sysHostCpuUser** (.1.3.6.1.4.1.3375.2.1.7.2.2.1.3)
- **sysHostCpuNice** (.1.3.6.1.4.1.3375.2.1.7.2.2.1.4)
- **sysHostCpuIdle** (.1.3.6.1.4.1.3375.2.1.7.2.2.1.5)
- **sysHostCpuSystem** (.1.3.6.1.4.1.3375.2.1.7.2.2.1.6)
- **sysHostCpuIrq** (.1.3.6.1.4.1.3375.2.1.7.2.2.1.7)
- **sysHostCpuSoftirq** (.1.3.6.1.4.1.3375.2.1.7.2.2.1.8)
- **sysHostCpuIowait** (.1.3.6.1.4.1.3375.2.1.7.2.2.1.9)

*Note: For each OID, perform the polls approximately ten seconds apart.*

2. For each OID, calculate the delta of the values from the two polls, as shown in the following formulas. Note that in the formulas shown, values such as **sysHostCpuUser2** and **sysHostCpuUser1** represent the values that result from the two polls you performed in step 1 for that OID.

```
DeltaCpuUser = sysHostCpuUser2 - sysHostCpuUser1
DeltaCpuNice = sysHostCpuNice2 - sysHostCpuNice1
DeltaCpuSystem = sysHostCpuSystem2 - sysHostCpuSystem1
DeltaCpuIdle = sysHostCpuIdle2 - sysHostCpuIdle1
DeltaCpuIrq = sysHostCpuIrq2 - sysHostCpuIrq1
DeltaCpuSoftirq = sysHostCpuSoftirq2 - sysHostCpuSoftirq1
DeltaCpuIowait = sysHostCpuIowait2 - sysHostCpuIowait1
```

3. Using the resulting delta values (for example, **DeltaCpuUser**), calculate the CPU[0-n] metric, according to the formula shown in table 13.9.

**To calculate the TMM CPU Usage metric**

1. Perform two separate polls of each of the following OIDs:

- **sysStatTmTotalCycles** (.1.3.6.1.4.1.3375.2.1.1.2.1.41)
- **sysStatTmIdleCycles** (.1.3.6.1.4.1.3375.2.1.1.2.1.42)
- **sysStatTmSleepCycles** (.1.3.6.1.4.1.3375.2.1.1.2.1.43)

*Note: For each OID, perform the polls approximately ten seconds apart.*

- For each OID, calculate the delta of the values from the two polls, as shown in the following example. Note that in the formula shown, values such as **sysStatTmTotalCycles2** and **sysStatTmTotalCycles1** represent the values that result from the two polls you performed in step 1 for each OID.

```
DeltaTmTotalCycles = sysStatTmTotalCycles2 -
sysStatTmTotalCycles1
```

```
DeltaTmIdleCycles = sysStatTmIdleCycles2 -
sysStatTmIdleCycles1
```

```
DeltaTmSleepCycles = sysStatTmSleepCycles2 -
sysStatTmSleepCycles1
```

- Using the resulting delta values (for example, **DeltaTmTotalCycles**), calculate the TMM CPU Usage metric, according to the formula shown in table 13.9.

## Collecting data on active sessions

You can use SNMP commands with an OID to gather data on active sessions. Table 13.10 shows the OID that you need to specify to gather data on active sessions.

Performance Graph (Configuration utility)	Graph Metrics	Required SNMP OIDs and the required calculations
Active Sessions	Established	apmAccessStatCurrentActiveSessions (.1.3.6.1.4.1.3375.2.6.1.4.3)

**Table 13.10** Required OIDs for collecting metrics on active sessions

## Collecting data on SSL transactions per second

You can use SNMP commands with an OID to gather data on SSL performance, in terms of transactions per second. Table 13.11 shows the OID that you need to specify to gather data on SSL TPS, along with the calculation that you must perform on the collected data.

Performance Graph (Configuration utility)	Graph Metrics	Required SNMP OIDs and the required calculations
SSL TPS	SSL TPS	sysStatClientTotConns (.1.3.6.1.4.1.3375.2.1.1.2.1.7) / time

**Table 13.11** Required OIDs for collecting metrics on SSL TPS

## Additional commands used for SNMP

You can use the following additional SNMP commands to view various statistics, including conducting a simple SNMP walk.

Task	Command
Performing an SNMP walk for SNMPv1	<code>snmpwalk -c &lt;communitystring&gt; -v &lt;1&gt; &lt;mgtlPofSecureAccessManager&gt; enterprises.3375.2.6</code>
Performing an SNMP walk for SNMPv2	<code>snmpwalk -c &lt;communitystring&gt; -v &lt;2c&gt; &lt;mgtlPofSecureAccessManager&gt; enterprises.3375.2.6</code>
Performing an SNMP walk for SNMPv3	<code>snmpwalk -v 3 -u &lt;username&gt; -a MD5 -A &lt;authPassword&gt; enterprises.3375.2.6</code> or <code>snmpwalk -v 3 &lt;username&gt; -x DES -X &lt;privacy password&gt; &lt;mgtlPofSecureAccessManager&gt; enterprises.3375.2.6</code>
Viewing global access statistics for SNMPv1	<code>snmpwalk -c &lt;communitystring&gt; -v &lt;1&gt; &lt;mgtlPofSecureAccessManager&gt; enterprises.3375.2.6.1.2</code>
Viewing global access statistics for SNMPv2	<code>snmpwalk -c &lt;communitystring&gt; -v &lt;2c&gt; &lt;mgtlPofSecureAccessManager&gt; enterprises.3375.2.6.1.2</code>
Viewing global access statistics for SNMPv3	<code>snmpwalk -v 3 -u &lt;username&gt; -a MD5 -A &lt;authPassword&gt; enterprises.3375.2.6.1.2</code> or <code>snmpwalk -v 3 &lt;username&gt; -x DES -X &lt;privacy password&gt; &lt;mgtlPofSecureAccessManager&gt; enterprises.3375.2.6.1.2</code>
Viewing global PPP statistics for SNMPv1	<code>snmpwalk -c &lt;communitystring&gt; -v &lt;1&gt; &lt;mgtlPofSecureAccessManager&gt; enterprises.3375.2.6.2.1</code>
Viewing global PPP statistics for SNMPv2	<code>snmpwalk -c &lt;communitystring&gt; -v &lt;2c&gt; &lt;mgtlPofSecureAccessManager&gt; enterprises.3375.2.6.2.1</code>
Viewing global PPP statistics for SNMPv3	<code>snmpwalk -v 3 -u &lt;username&gt; -a MD5 -A &lt;authPassword&gt; enterprises.3375.2.6.2.1</code> or <code>snmpwalk -v 3 &lt;username&gt; -x DES -X &lt;privacy password&gt; &lt;mgtlPofSecureAccessManager&gt; enterprises.3375.2.6.2.1</code>
Viewing profile access statistics for SNMPv1	<code>snmpwalk -c &lt;communitystring&gt; -v &lt;1&gt; &lt;mgtlPofSecureAccessManager&gt; enterprises.3375.2.6.1.1</code>
Viewing profile access statistics for SNMPv2	<code>snmpwalk -c &lt;communitystring&gt; -v &lt;2c&gt; &lt;mgtlPofSecureAccessManager&gt; enterprises.3375.2.6.1.1</code>
Viewing profile access statistics for SNMPv3	<code>snmpwalk -v 3 -u &lt;username&gt; -a MD5 -A &lt;authPassword&gt; enterprises.3375.2.6.1.1</code> or <code>snmpwalk -v 3 &lt;username&gt; -x DES -X &lt;privacy password&gt; &lt;mgtlPofSecureAccessManager&gt; enterprises.3375.2.6.1.1</code>

**Table 13.12** Additional commands to view SNMP statistics



# 14

---

## Session Variables

---

- Introducing session variables
- Introducing Tcl
- Session variables reference
- Understanding network access resource variable attributes
- Using session variables in the Configuration utility





## Introducing session variables

The rules in an access policy store the values that the actions return in session variables. A session variable contains a number or string that represents a specific piece of information.

You can use the session variable strings in the visual policy editor, to customize a rule for a specific action in an access policy. For more information on configuring access policy rules with session variables, see *Assigning variables*, on page 6-10, and *Using advanced access policy rules*, on page 11-16.

When you use session variables, you typically write them in custom rules, in the Tcl language, or you use them in the variable assign action.

This appendix includes three tables.

- Table 14.1, *Session variables for BIG-IP Access Policy Manager*, contains the session variables returned by access policy actions.
- Table 14.2, *Special purpose user session variables*, contains special purpose session variables that provide functions in a user session, but are not returned by specific access policy actions.
- Table 14.3, *Network access resource configuration variables and attributes*, contains the session variables generated by a network access resource, and the formats of those variables, for use with the variable assign action.

---

### ◆ Note

*When using session variables in an access policy configuration, for example, in a logging agent, a session variable might or might not exist depending on the result of the access policy process.*

## Introducing Tcl

You write rules in Tcl. Although this appendix is not an exhaustive reference for writing and using Tcl expressions, it includes some common operators and syntax rules. Tcl expressions begin with the syntax **expr**. For more information, see <http://www.tcl.tk/man/tcl8.5/TclCmd/expr.htm>.

---

### ◆ Note

*You use iRules® on the BIG-IP® system to provide functionality to the BIG-IP system components. Tcl commands specific to iRules are not available in access policy rules.*

## Standard operators

You can use Tcl standard operators with most BIG-IP® Access Policy Manager® rules. You can find a full list of these operators in the Tcl online manual, at <http://www.tcl.tk/man/tcl8.5/TclCmd/expr.htm>.

Standard operators include:

- **- + ~ !**  
Unary minus, unary plus, bit-wise NOT, logical NOT. None of these operators may be applied to string operands, and bit-wise NOT may be applied only to integers.
- **\*\***  
Exponentiation. Valid for any numeric operands.
- **\* / %**  
Multiply, divide, remainder. None of these operators may be applied to string operands, and remainder may be applied only to integers. The remainder will always have the same sign as the divisor and an absolute value smaller than the divisor.
- **+ -**  
Add and subtract. Valid for any numeric operands.
- **<< >>**  
Left and right shift. Valid for integer operands only. A right shift always propagates the sign bit.
- **< > <= >=**  
Boolean less than, greater than, less than or equal to, and greater than or equal to. Each operator produces 1 if the condition is true, 0 otherwise. These operators may be applied to strings as well as numeric operands, in which case string comparison is used.
- **== !=**  
Boolean equal to and not equal to. Each operator produces a zero/one result. Valid for all operand types.
- **eq ne**  
Boolean string equal to and string not equal to. Each operator produces a zero/one result. The operand types are interpreted only as strings.

- **in ni**  
List containment and negated list containment. Each operator produces a zero/one result and treats its first argument as a string and its second argument as a Tcl list. The in operator indicates whether the first argument is a member of the second argument list; the ni operator inverts the sense of the result.
- **&**  
Bit-wise AND. Valid for integer operands only.
- **^**  
Bit-wise exclusive OR. Valid for integer operands only.
- **|**  
Bit-wise OR. Valid for integer operands only.
- **&&**  
Logical AND. Produces a 1 result if both operands are non-zero, 0 otherwise. Valid for boolean and numeric (integers or floating-point) operands only.
- **||**  
Logical OR. Produces a 0 result if both operands are zero, 1 otherwise. Valid for boolean and numeric (integers or floating-point) operands only.
- **x?y:z**  
If-then-else, as in C. If x evaluates to non-zero, then the result is the value of y. Otherwise the result is the value of z. The x operand must have a boolean or numeric value.

## Rule operators

A rule operator compares two operands in an expression. In addition to using the Tcl standard operators, you can use the operators listed below.

- **contains** - Tests if one string contains another string.
- **ends\_with** - Tests if one string ends with another string.
- **equals** - Tests if one string equals another string.
- **matches** - Tests if one string matches another string.
- **matches\_regex** - Tests if one string matches a regular expression.
- **starts\_with** - Tests if one string starts\_with another string.
- **switch** - Evaluates one of several scripts, depending on a given value.

## Logical operators

Logical operators are used to compare two values.

- **and** - Performs a logical “and” comparison between two values.
- **not** - Performs a logical “not” action on a value.
- **or** - Performs a logical “or” comparison between two values.

## Session variables reference

This table includes session variables and related reference information for each session variable that you can use with Access Policy Manager.

For a set of special purpose session variables, see Table 14.2

Agent	Name	Type	Format	Description
Active Directory action	session.ad.\$name.queryresult	bool		Result of the Active Directory query. 0 - Failed 1 - Passed
	session.ad.\$name.authresult	bool		Result of the Active Directory authentication attempt. 0 - Failed 1 - Passed
	session.ad.\$name.attr.\$attr_name	string		Users attributes retrieved during Active Directory query. Each attribute is converted to a separate session variable.
	session.ad.\$name.attr.group.\$attr_name	string		User's group attributes retrieved during Active Directory query. Each group attribute is converted to a separate session variable.
LDAP action	session.ldap.\$name.authresult	bool		Result of the LDAP authentication attempt. 0 - Failed 1 - Passed
	session.ldap.\$name.attr.\$attr_name	string		Users attributes retrieved during AD query. Each attribute is converted to a separate session variable.
	session.ldap.\$name.queryresult	bool		Result of the LDAP query. 0 - Failed 1 - Passed
RADIUS action	session.radius.\$name.authresult	bool		Result of the RADIUS authentication attempt. 0 - Failed 1 - Passed

**Table 14.1** Session variables for BIG-IP Access Policy Manager

Agent	Name	Type	Format	Description
RADIUS action	session.radius.\$name.attr. \$attr_name	string		User attributes retrieved during RADIUS authentication. Each attribute is converted to a separate session variable.
Denied Ending	session.policy.result	string	"access_denied"	The result of the access policy. The result is the ending; for this ending, the result is <b>access_denied</b> .
Redirect Ending	session.policy.result	string	"redirect"	The result of the access policy. The result is the ending; for this ending, the result is <b>redirect</b> .
	session.policy.result.redirect.url	string		The URL specified in the redirect, for example, "http://www.siterequest.com"
Allowed Ending	session.policy.result	string	"allowed"	The result of the access policy. The result is the ending; for this ending, the result is <b>allowed</b> .
	session.policy.result.webtop. network_access.autolaunch	string	"resname"	The resource that is automatically started for a network access webtop
	session.policy.result.webtop.type	string	"network_access"	The type of webtop resource. The webtop type can be <b>network_access</b> or <b>web_application</b> .
Advanced Resource Assign	session.assigned.bwc.dynamic	string		The assigned dynamic bandwidth control policy
	session.assigned.bwc.static	string		The assigned static bandwidth control policy
Decision box	session.decision_box.last.result	integer		0 - User chooses option 2 on the decision page, which corresponds to the fallback rule branch in the action 1 - User chooses option 1 on the decision page
File check	session.windows_check_file. \$name.item_0.exist	string		True - if all files exist on the client.

**Table 14.1** Session variables for BIG-IP Access Policy Manager

Agent	Name	Type	Format	Description
File check	session.windows_check_file. \$name.item_0.result	integer		Set when files on the client meet the configured attributes.
	session.windows_check_file. \$name.item_0.md5	string		MD5 value of a checked file.
	session.windows_check_file. \$name.item_0.version	string		The version of a checked file.
	session.windows_check_file. \$name.item_0.size	integer		The file size, in bytes.
	session.windows_check_file. \$name.item_0.modified			Date the file was modified in UTC form.
	session.windows_check_file. \$name.item_0.signer			File signer information.
Logon Page (CAPTCHA challenge)	session.logon.captcha.tracking	unsigned integer		The unsigned integer is treated as a bitmask. Determines whether to track successful/unsuccessful logon attempts by IP (bit in 0 position) and/or by username (bit in 1 position) when CAPTCHA is enabled. Should not be used by external modules because it is intended for very specific purposes.
Machine Cert Auth	session.check_machinecert.last.result			<p>0 - Neither certificate nor private key were found.</p> <p>1 - Both certificate and private key were found.</p> <p>2 - Certificate was found, but private key was not found.</p> <p>-2 - Various errors, such as:</p> <ul style="list-style-type: none"> <li>• Nothing received from client.</li> <li>• Data received is not in correct format.</li> <li>• Linux client is trying to access the agent; (Machine Cert Auth is not supported on Linux)</li> <li>• Incorrect configuration. For example if CA profile is not configured.</li> </ul>

**Table 14.1** Session variables for BIG-IP Access Policy Manager

Agent	Name	Type	Format	Description
OTP Generate	session.otp.assigned.val	string		Generated one-time password value to send to the end user. Example message: One-Time Passcode: %{session.otp.assigned.val}
	session.otp.assigned.expire	string		Internally used timestamp; OTP expiration in seconds since this date and time: (00:00:00 UTC, January 1, 1970)
	session.otp.assigned.ttl	string		OTP time-to-live; configurable as OTP timeout in seconds. Example message: OTP expires after use or in %{session.otp.assigned.ttl} seconds
OTP Verify	session.otp.verify.last.authresult	bool		Result of OTP authentication attempt: 0 - Failed 1 - Passed
Process check	session.windows_check_process.\$name.result	integer		0 - Failure 1 - Success -1 - Invalid check expression
Windows Registry check	session.windows_check_registrys.\$name.result	integer		0 - Failure 1 - Success -1 - Invalid check expression
Windows info	session.windows_info_os.\$name.ie_version	string		Stores the Internet Explorer version
	session.windows_info_os.\$name.ie_updates	string	"!SP2!KB12345!KB54321!"	A list of installed SP and KB fixes for Internet Explorer
	session.windows_info_os.\$name.platform	string		Win7 - Windows 7 Win8 - Windows 8 WinVI - Windows WinXP - Windows XP Win2003 - Windows 2003 Server WinLH - Windows 2008

**Table 14.1** Session variables for BIG-IP Access Policy Manager

Agent	Name	Type	Format	Description
Window info	session.windows_info_os.\$name.updates	string	"!SP2!KB12345!KB54321!"	A list of installed SP and KB fixes for Windows
	session.windows_info_os.\$name.user	string		List of current windows user names
	session.windows_info_os.\$name.computer	string		List of computer names
Resource allocation	session.assigned.resources	string	"resource1 resource2"	A space-delimited list of assigned resources.
	session.assigned.webtop	string	'webtop_name'	The name of the assigned webtop.
Client certificate authentication	session.ssl.cert.x509extension	string		X509 extensions
	session.ssl.cert.valid	string		Certificate Result (OK or error string)
	session.ssl.cert.exist	integer		0 - certificate does not exist 1- certificate exists
	session.ssl.cert.version	string		Certificate version
	session.ssl.cert.subject	string		Certificate subject field
	session.ssl.cert.serial	string		Certificate serial number
	session.ssl.cert.end	string		Validity end date
	session.ssl.cert.start	string		Validity start date
	session.ssl.cert.issuer	string		Certificate issuer
	session.ssl.cert.whole	string		The whole certificate
Session management	session.ui.mode	enum		The UI mode, as determined by HTTP headers.
	session.ui.lang	string	"en"	The language in use in the session.
	session.ui.charset		"	The character set used in the session.
	session.client.type	enum	portalclient "Standalone"	The client type as determined by HTTP headers.

**Table 14.1** Session variables for BIG-IP Access Policy Manager



Agent	Name	Type	Format	Description
Session management	session.client.version	string		
	session.client.js	bool		
	session.clientactivex	bool		
	session.client.plugin	bool		
	session.client.platform	string	WinNT "Win2k" "WinXP" "WinVI" "Linux" "MacOS" iOS Android	The client platform as determined by HTTP headers.
	session.user.access_mode	string	"local"	Enables direct access to a resource from the webtop. Used with Citrix resources. See <i>Using the session.user.access_mode session variable</i> , on page 14-10.

**Table 14.1** Session variables for BIG-IP Access Policy Manager

## Using the session.user.access\_mode session variable

The **session.user.access\_mode** session variable allows users to be assigned a full webtop for publishing resources with direct access to the resources, without an Access Policy Manager resource assigned. This is designed to simplify access to internal resources for local users.

The valid values for this session variable are:

- **local** - enables local access mode
- **Any other value** - disables local access mode. This access mode is disabled in a session by default.

You can use this variable with Citrix resources only.

### Configuring session.user.access\_mode

1. In the Visual Policy Editor, on an access policy branch, add a **Variable Assign** agent with the custom expression  

```
"session.user.access_mode" = expr { "local" }
```
2. Assign a Citrix resource and full webtop on the same access policy branch.  
The resource connection is made directly to the Citrix server.

## Special purpose user session variables

Use the following session variables with the variable assign action to customize the behavior of a user session.

Name	Type	Format	Description
session.assigned.acls	string	"ACL1 ACL3 ACL5"	A space-delimited list of assigned ACLs.
session.assigned.acls.sorted	string	"ACL1 ACL3 ACL5"	A space-delimited list of assigned ACLs. This variable is created to store the list of ACLs. To modify the list of ACLs with the variable assign action or an advanced access policy rule, modify the previous session variable, <b>session.assigned.acls</b> .
session.assigned.clientip	string	xxx.xxx.xxx.xxx For example, 192.168.12.10	The informational variable that stores the client IP address assigned by Access Policy Manager after the access policy completes.
session.end	string	admin_terminated logged_out timed_out	An informational variable that stores the reason the session was terminated.
session.assigned.leasepool	string	lp1	The lease pool assigned to the client session.
session.assigned.resources	string	"res1 res3 res5"	A space-delimited list of assigned resource names. This list is generated based on the list of assigned resource groups.
session.assigned.route_domain	int	1	The route domain ID number assigned to the client session.
session.user.sessionid	string	string (8 hex characters)(	The ID for a session. For example, <b>e2d0b683</b> .
session.logon.last.username	string	"username"	You can use the session user name variable with the variable assign action to replace the user name value that is passed to an authentication action in the access policy. An authentication action then authenticates with this user name value.
session.logon.last.password	string	"password"	The session password variable contains the user password that is collected in the logon page action. This variable stores the password, then sends it to the authentication server. You should not configure the variable assign action to replace this variable.

**Table 14.2** Special purpose user session variables

## Understanding network access resource variable attributes

This table includes the variables you can access in a network access resource, and the formats and values of the variable attributes.

Use this table with the variable assign action, to correctly format the replacement attribute for an existing network access resource configuration variable.

When the session variable requires that you write replacement XML in a specific format, the XML is presented in this table as **<tag>tagdata</tag>**. In this example, you type both the opening **<tag>** and the closing **</tag>** elements as provided, then type the actual XML data between the opening and closing elements. For example, the following is an entry in the table.

```
<dns>
<dns_primary>IP Address</ dns_primary>
<dns_secondary>IP Address</ dns_secondary>
</dns>
```

**Figure 14.1** Network access resource XML formatting example

The following is an example of replacement code you could write, based on this table entry.

```
<dns>
<dns_primary>4.2.2.1</ dns_primary>
<dns_secondary>4.2.2.2</ dns_secondary>
</dns>
```

**Figure 14.2** Network access resource XML formatting example

### ◆ Important

*The result of an evaluated expression or custom expression that you use to replace a network access property must provide a value in the format described in the Attribute value format column.*

Network access resource property	Type	Attribute value format
leasepool_name	string	The attribute value is the name of a leasepool that exists on Access Policy Manager
snat_type	integer	The attribute value is <b>0</b> , <b>2</b> , or <b>3</b> . 0 - None (no SNAT) 2 - SNAT pool (assigned with the variable <b>snatpool_name</b> ) 3 - Automap

**Table 14.3** Network access resource configuration variables and attributes

Network access resource property	Type	Attribute value format
snatpool_name	string	The attribute value is the name of an SNAT pool. The SNAT pool must be configured on the Access Policy Manager.
compression	int	The attribute value is <b>0</b> or <b>1</b> . 0 = disable compression 1 = enable compression
client_proxy_settings	Bool String IPAddress Number Bool Vector(String) (see example)	The attribute is XML, formatted as follows: <pre>&lt; client_proxy_settings &gt; &lt;client_proxy&gt;1&lt;/client_proxy&gt; &lt;client_proxy_script&gt;proxy_script &lt;/client_proxy_script&gt; &lt;client_proxy_address&gt;proxyaddress &lt;/ client_proxy_address&gt; &lt;client_proxy_port&gt;proxyport&lt;/client_proxy_port&gt; &lt;client_proxy_local_bypass&gt;1 &lt;/client_proxy_local_bypass&gt; &lt;client_proxy_exclusion_list&gt; &lt;item&gt;exclusion_list_item1&lt;/item&gt; &lt;item&gt;exclusion_list_item2&lt;/item&gt; &lt;/client_proxy_exclusion_list&gt; &lt;/client_proxy_settings&gt;</pre> <p>Note that <b>&lt;client_proxy&gt;</b> should have the value <b>1</b> for the other settings to be effective, otherwise all other setting from <b>&lt;client_proxy_settings&gt;</b> will be ignored.</p>
drive_mapping	Vector (Struct)	The attribute is XML, formatted as follows: <pre>&lt;drive_mapping&gt; &lt;item&gt; &lt;description&gt; description&lt;/description&gt; &lt;path&gt;drive_path&lt;/path&gt; &lt;drive&gt;drive_letter&lt;/drive&gt; &lt;/item&gt; &lt;/drive_mapping&gt;</pre> <p>Note that the drive letter range is from D to Z.</p>
session_update_threshold	int	The attribute value is the session update threshold, in seconds.
session_update_window	int	The attribute value is the session update window, in seconds.
address_space_include_dns_name	Vector (string)	The attribute is XML, formatted as follows: <pre>&lt;address_space_include_dns_name&gt; &lt;item&gt;&lt;dnsname&gt; dnsname1 &lt;/dnsname&gt;&lt;/item&gt; &lt;item&gt;&lt;dnsname&gt; dnsname2 &lt;/dnsname&gt;&lt;/item&gt; &lt;/address_space_include_dns_name&gt;</pre>

**Table 14.3** Network access resource configuration variables and attributes

Network access resource property	Type	Attribute value format
address_space_include_subnet	Vector (network)	The attribute value is a space-separated list of subnets. For example: <b>192.168.30.0/255.255.255.0</b> <b>172.30.11.0/255.255.255.0</b>
address_space_exclude_subnet	Vector(network)	The attribute value is a space-separated list of subnets. For example: <b>192.168.30.0/255.255.255.0</b> <b>172.30.11.0/255.255.255.0</b>
address_space_protect	Bool	The attribute value is <b>0</b> or <b>1</b> . 0 = disable address space protection 1 = enable address space protection
address_space_local_subnets_excluded	Bool	The attribute value is <b>0</b> or <b>1</b> . 0 = disable address space local subnet exclusion 1 = enable address space local subnet exclusion
address_space_dhcp_requests_excluded	Bool	The attribute value is <b>0</b> or <b>1</b> . 0 = disable address space DHCP request exclusion 1 = enable address space DHCP request exclusion
split_tunneling	Bool	The attribute value is <b>0</b> or <b>1</b> . 0 = disable split tunneling 1 = enable split tunneling <b>Note:</b> If <i>split_tunneling</i> is set to <b>0</b> then you must set the following variables: <b>address_space_exclude_subnet = ""</b> <b>address_space_include_subnet = "128.0.0.0/128.0.0.0 0.0.0.0/128.0.0.0"</b> <b>address_space_include_dns_name = ""</b>
dns	String	The attribute is XML, formatted as follows: <dns> <dns_primary>IPAddress</ dns_primary> <dns_secondary>IPAddress</ dns_secondary> </dns>
dns_suffix	String	The DNS Default Domain Suffix. For example, <b>siterequest.com</b> .
wins	String	The attribute is XML, formatted as follows: <wins> <wins_primary >IPAddress</ wins_primary > <wins_secondary>IPAddress</ wins_secondary> </wins>

**Table 14.3** Network access resource configuration variables and attributes

Network access resource property	Type	Attribute value format
static_host	Vector(staticHost)	The attribute is XML, formatted as follows: <pre>&lt;static_host&gt; &lt;item&gt; &lt;hostname&gt;hostname&lt;/hostname&gt; &lt;address&gt;IPAddress&lt;/address&gt; &lt;/item&gt; &lt;/static_host&gt;</pre>
client_interface_speed	int	The number for the client interface speed value in the network access resource, in bytes.
client_ip_filter_engine	Bool	The attribute value is <b>0</b> or <b>1</b> . 0 = disable integrated IP filtering engine 1 = enable integrated IP filtering engine
client_power_management	Bool	The attribute value is <b>0</b> or <b>1</b> . 0 = disable client power management 1 = enable client power management
microsoft_network_client	Bool	The attribute value is <b>0</b> or <b>1</b> . 0 = disable the Client for Microsoft Networks option 1 = enable the Client for Microsoft Networks option
microsoft_network_server	Bool	The attribute value is <b>0</b> or <b>1</b> . 0 = disable the File and printer sharing for Microsoft Networks option 1 = enable the File and printer sharing for Microsoft Networks option
warn_before_application_launch	Bool	The attribute value is <b>0</b> or <b>1</b> . 0 = disable the <b>Display warning before launching applications</b> option 1 = enable the <b>Display warning before launching applications</b> option
application_launch	Vector(AppLaunch)	The attribute is XML, formatted as follows: <pre>&lt;application_launch&gt; &lt;item&gt; &lt;path&gt;path&lt;/path&gt; &lt;parameter&gt;string&lt;/parameter&gt; &lt;os_type&gt;os_type&lt;/os_type&gt; &lt;/item&gt; &lt;/application_launch&gt;</pre> For the <b>&lt;os_type&gt;</b> value, type WINDOWS, MAC, or IOS. This field is case sensitive.

**Table 14.3** Network access resource configuration variables and attributes

Network access resource property	Type	Attribute value format
provide_client_cert	Bool	The attribute value is <b>0</b> or <b>1</b> . 0 = disable the <b>Provide client certificate on Network Access connection when requested</b> option 1 = enable the <b>Provide client certificate on Network Access connection when requested</b> option
tunnel_port_dtls	int	The attribute is the DTLS port, for example <b>4433</b> . <b>Note:</b> setting this to any number other than <b>0</b> enables DTLS in the network access resource, and sets the number you specify as the DTLS port.

**Table 14.3** Network access resource configuration variables and attributes

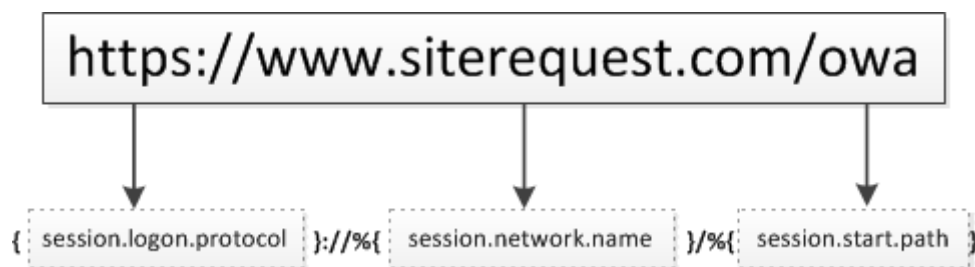


## Using session variables in the Configuration utility

With BIG-IP® Access Policy Manager® Configuration utility, for many configuration fields, you can use a session variable to retrieve data from the session that populates a field at session runtime.

Session variables can be used with the following configuration notes:

- Not all fields support session variables. See *Supported fields for session variables in the Configuration utility*, on page 14-17 for supported fields and configuration information.
- Session variables in a configuration field can be encrypted.
- Any field that supports session variables can support multiple session variables. For example, you can create a URL in a field from session variables.
  - Use **session.logon.protocol** to specify the protocol for the URL.
  - Use **session.network.name** to specify the host address portion of the URL.
  - Use **session.start.path** to specify the path info that follows the host address.



**Figure 14.3** Multiple session variables used to specify a URL.

## Supported fields for session variables in the Configuration utility

The following fields in the configuration utility are compatible with session variables.

- *Redirect URL field session variable*, on page 14-18
- *Start URI session variable for portal access webtop*, on page 14-19
- *Form-based SSO configuration session variables*, on page 14-20
- *NTLM v1 and v2 SSO configuration session variables*, on page 14-23
- *HTTP Basic SSO configuration session variables*, on page 14-24
- *Kerberos SSO configuration session variables*, on page 14-25
- *SSO header insertion session variables*, on page 14-26

- *LDAP Query search filter session variable*, on page 14-27
- *AD Query search filter session variable*, on page 14-28
- *AAA server form-based hidden parameter session variable*, on page 14-29
- *Default customization parameters for Network Access*, on page 14-30
- *Network access launch application session variables*, on page 14-31
- *Network access drive mapping session variable*, on page 14-32
- *Application tunnel resource session variables*, on page 14-33
- *Remote desktop resource session variables*, on page 14-34

## Redirect URL field session variable

The Redirect URL field appears when you add or edit an ending in the visual policy editor. You use session variables in this field to dynamically generate the Redirect URL for the client.

### Example

In this example, when the user reaches a redirect ending in the access policy, the access policy creates the Redirect URI dynamically from the current LDAP session.

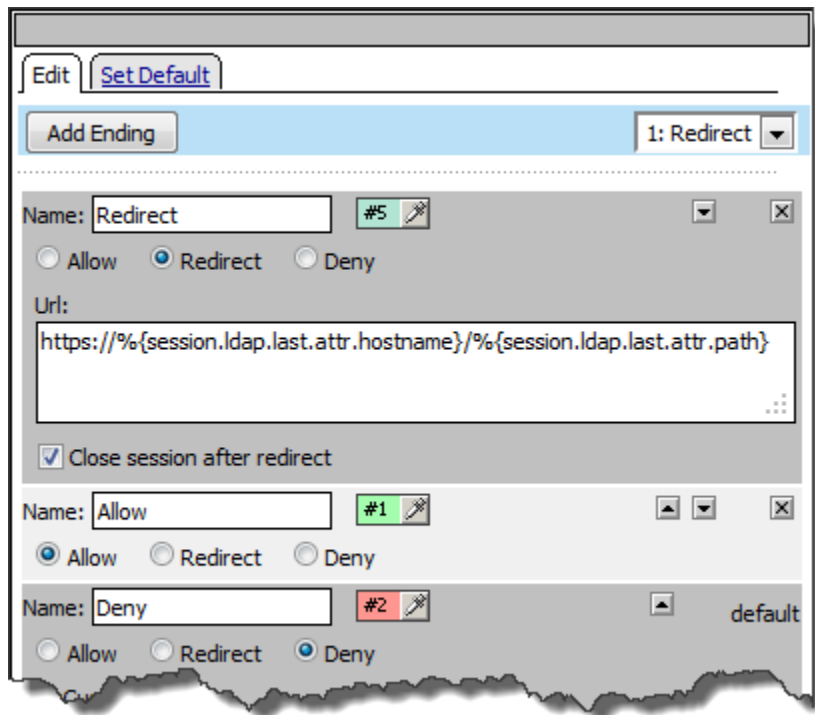
```
https://{%session.ldap.last.attr.hostname}/{%session.ldap.last.attr.path}
```

**Figure 14.4** Session variables using LDAP session data

---

#### ◆ Note

*Currently the Redirect URL field only supports session variables for the host and path, and not for scheme.*



*Figure 14.5 LDAP session variables in Redirect URL field*

## Start URI session variable for portal access webtop

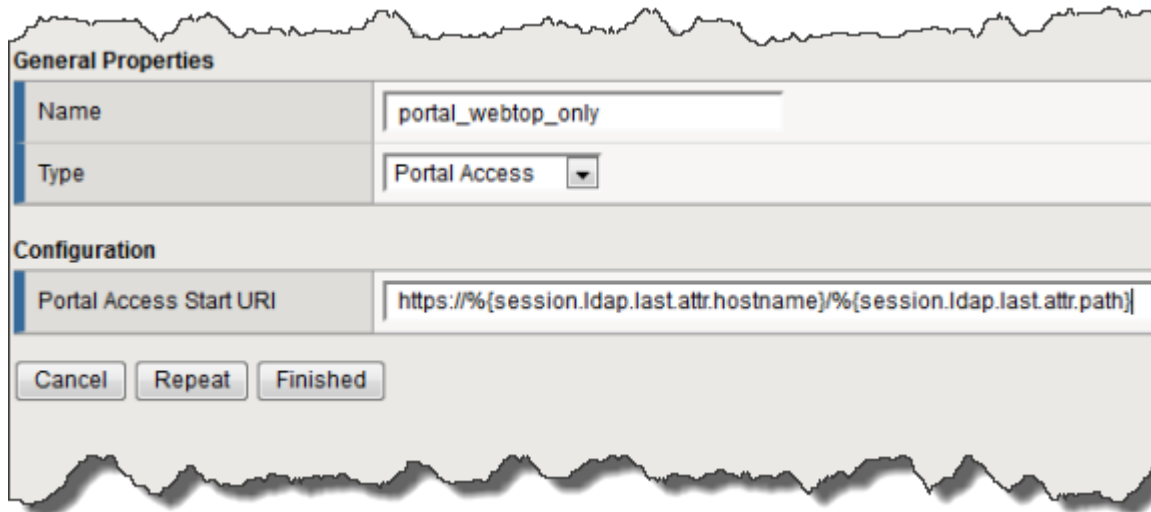
The Start URI field for a portal access webtop appears when you create a webtop for portal access only. You use session variables in this field to dynamically generate the webtop URL for the client.

### Example

In this example, when the user reaches the portal access webtop, the web application starts at the URI that the access policy dynamically generates based on variables from the current LDAP session.

```
https://%{session.ldap.last.attr.hostname}/%{session.ldap.last.attr.path}
```

*Figure 14.6 LDAP session variables for host and path*



The screenshot shows a configuration window with a torn-paper border. It is divided into two main sections: 'General Properties' and 'Configuration'. In the 'General Properties' section, the 'Name' field contains 'portal\_webtop\_only' and the 'Type' dropdown is set to 'Portal Access'. The 'Configuration' section has a 'Portal Access Start URI' field containing the URL 'https://[%{session.ldap.last.attr.hostname}]/[%{session.ldap.last.attr.path}]'. At the bottom of the window are three buttons: 'Cancel', 'Repeat', and 'Finished'.

*Figure 14.7 LDAP session variables in Portal Access Start URI field*

## Form-based SSO configuration session variables

There are several fields you can configure with session variables in the Form-based SSO creation page:

- **Username Source** field in the Credentials Source area (by default populated with `session.sso.token.last.username`)
- **Password Source** field in the Credentials Source area (by default populated with `session.sso.token.last.password`)
- **Start URI** field in the SSO Method Configuration area
- **Form Action** field in the SSO Method Configuration area
- **Form Parameter for User Name** field in the SSO Method Configuration area
- **Form Parameter for Password** field in the SSO Method Configuration area
- **Hidden Form Parameters/Values** field in the SSO Method Configuration area
- **Successful Logon Detection Match Value** field in the SSO Method Configuration area

## Example

In this example, SSO information to maintain the user session is determined at runtime by SSO session variables. The following session variables are used to populate the SSO form.

```
session.sso.token.last.username  
session.sso.token.last.password  
session.sso.start.uri  
session.sso.myform.action  
session.sso.myform.userparamname  
session.sso.hidden_type  
session.sso.result_match
```

*Figure 14.8 SSO session variables for SSO form*

General Properties: Basic	
Name	
SSO Method	Forms
Use SSO Template	None
Credentials Source	
Username Source	session.sso.token.last.username
Password Source	session.sso.token.last.password
SSO Method Configuration	
Start URI	<code>\${session.sso.starturi}</code>
Pass Through	<input type="checkbox"/> Enable
Form Method	POST
Form Action	<code>%(session.sso.myform.action)</code>
Form Parameter For User Name	<code>%(session.sso.myform.userparamname)</code>
Form Parameter For Password	<code>%(session.sso.userparamname)</code>
Hidden Form Parameters/Values	<code>LoginType \${session.sso.hidden_type}</code>
Successful Logon Detection Match Type	By Resulting Redirect URL
Successful Logon Detection Match Value	<code>\${session.sso.result_match}</code>

*Figure 14.9 SSO session variables for SSO fields*

## NTLM v1 and v2 SSO configuration session variables

There are several fields you can configure with session variables on the NTLM SSO creation page:

- **Username Source** field in the Credentials Source area (by default populated with `session.sso.token.last.username`)
- **Password Source** field in the Credentials Source area (by default populated with `session.sso.token.last.password`)
- **Domain Source** field in the Credentials Source area (by default populated with `session.logon.last.domain`)
- **NTLM Domain** field in the SSO Method Configuration area

### Example

In this example, SSO information to establish the user session is determined at runtime by SSO session variables. The following session variables are used to populate the SSO form.

```
session.sso.token.last.username  
session.sso.token.last.password  
session.sso.start.uri  
session.logon.last.domain
```

*Figure 14.10 SSO session variables for NTLM configuration*

**General Properties:** Basic ▼

Name	sso_NTLM1
Partition / Path	Common
SSO Method	NTLMV1

**Credentials Source**

Username Source	session.sso.token.last.username
Password Source	session.sso.token.last.password
Domain Source	session.logon.last.domain

**SSO Method Configuration**

Username Conversion	<input type="checkbox"/> Enable
NTLM Domain	MYDOMAIN

Update Delete

*Figure 14.11 SSO session variables for NTLM fields*

## HTTP Basic SSO configuration session variables

There are two fields automatically configured with session variables in the HTTP Basic SSO creation page:

- **Username Source** field in the Credentials Source area (by default populated with `session.sso.token.last.username`)
- **Password Source** field in the Credentials Source area (by default populated with `session.sso.token.last.password`)



## Example

In this example, SSO information to establish the user session is determined at runtime by SSO session variables. The following session variables are automatically populated in the SSO form.

```
session.sso.token.last.username  
session.sso.token.last.password
```

*Figure 14.12 SSO session variables for HTTP Basic SSO form*

The screenshot shows a configuration window titled 'General Properties: Basic'. It contains several sections: 'Name' with the value 'HTTP\_Basic\_SSO', 'SSO Method' with the value 'HTTP Basic', 'Credentials Source' with 'Username Source' set to 'session.sso.token.last.username' and 'Password Source' set to 'session.sso.token.last.password', and 'SSO Method Configuration' with 'Username Conversion' set to 'Enable'. At the bottom are 'Cancel' and 'Finished' buttons.

*Figure 14.13 SSO session variables for HTTP Basic SSO fields*

## Kerberos SSO configuration session variables

There are two fields automatically configured with session variables in the Kerberos SSO creation page:

- **Username Source** field in the Credentials Source area (by default populated with **session.sso.token.last.username**)
- **Password Source** field in the Credentials Source area (by default populated with **session.sso.token.last.password**)

## Example

In this example, SSO information to establish the user session is determined at runtime by SSO session variables. The following session variables are automatically populated in the SSO form.

```
session.sso.token.last.username  
session.sso.token.last.password
```

*Figure 14.14 SSO session variables for Kerberos SSO form*

The screenshot shows a web-based configuration form for SSO. At the top, there is a tab labeled 'General Properties:' with a dropdown menu set to 'Basic'. Below this, there are two rows of fields: 'Name' with the value 'Kerberos\_SSO' and 'SSO Method' with the value 'Kerberos'. The next section is titled 'Credentials Source' and contains two rows: 'Username Source' with the value 'session.sso.token.last.username' and 'User Realm Source' with the value 'session.logon.last.domain'. The final section is titled 'SSO Method Configuration' and contains one row: 'Kerberos Realm' with the value 'kbr'.

General Properties: Basic	
Name	Kerberos_SSO
SSO Method	Kerberos
Credentials Source	
Username Source	session.sso.token.last.username
User Realm Source	session.logon.last.domain
SSO Method Configuration	
Kerberos Realm	kbr

*Figure 14.15 SSO session variables for Kerberos SSO fields*

## SSO header insertion session variables

You can use session variables to specify SSO header name-value pairs. To set name-value pairs, you must select **Advanced** on any SSO configuration page.

## Example

In this example, SSO header insertion name-value pairs can be determined at runtime with session variables. Use custom session variables to populate these fields in the SSO form. The following session variables are examples only.

```
session.custom.header_name1
session.custom.header_value1
session.custom.header_name2
session.custom.header_value2
```

*Figure 14.16 SSO session variables for SSO header name-value pairs*

The screenshot shows a configuration window for SSO. At the top, there's a tab labeled 'General Properties:' with a dropdown menu set to 'Advanced'. Below this, there are two main sections: 'Name' and 'SSO Method'. The 'Name' field contains 'Kerberos\_SSO'. The 'SSO Method' field contains 'Kerberos'. Below these, there's a 'Headers' section. It contains two input fields: 'Name' with the value '%{session.custom.header\_name2}' and 'Value' with the value '%{session.custom.header\_value2}'. There is an 'Add' button below these fields. Below the 'Add' button, there's a list box containing the text '%{session.custom.header\_name1} : %{session.custom.header\_value1}'. At the bottom of the list box, there are 'Edit' and 'Delete' buttons.

*Figure 14.17 SSO session variables for SSO header insertion name-value pairs*

## LDAP Query search filter session variable

You can use a session variable for the LDAP Query SearchFilter parameter in the visual policy editor, to specify search parameters.

## Example

In this example, the SearchFilter field is populated with the username from the current session.

```
(sAmAccountName=%{session.logon.last.username})
```

**Figure 14.18** Session variable for LDAP Query

The screenshot shows a web-based configuration interface for an LDAP Query. At the top, there are two tabs: 'Properties\*' and 'Branch Rules'. Below the tabs, the 'Name' field is set to 'LDAP Query'. The main configuration area is titled 'LDAP' and contains several fields: 'Type' is set to 'Query' (with a dropdown arrow), 'Server' is set to 'None' (with a dropdown arrow), 'SearchDN' is an empty text field, 'SearchFilter' contains the LDAP query '(sAmAccountName=%{session.logon.last.username})', and 'Fetch Nested Groups' is set to 'Disabled' (with a dropdown arrow). The interface has a light gray background and a white border.

**Figure 14.19** SearchFilter session variable for LDAP Query

## AD Query search filter session variable

You can use a session variable for the AD Query SearchFilter parameter in the visual policy editor, to specify search parameters.

## Example

In this example, the SearchFilter field is populated with the Subject Alternative Name from the current Active Directory session.

```
(sAmAccountName=%{session.cert.last.subjAltName})
```

**Figure 14.20** Session variable for AD Query

Properties\* [Branch Rules](#)

Name:

**Active Directory**

Type	Query
Server	None
SearchFilter	(sAMAccountName={session.cert.last.subjAltName})
Fetch Primary Group	Disabled
Cross Domain Support	Disabled
Fetch Nested Groups	Disabled
Complexity check for Password	Disabled

[Reset](#)

*Figure 14.21 SearchFilter session variable for Active Directory query*

## AAA server form-based hidden parameter session variable

You can use a session variable in the AAA Server configuration Hidden Form Parameters/Values field, to specify hidden form parameters for AAA server form-based authentication.

### Example

In this example, the Hidden Form Parameters/Values field is populated with a hidden form submission command that submits the username associated with the session.

```
submit_form Submit%{session.logon.last.username}
```

*Figure 14.22 Session variable for AAA form-based authentication hidden parameters*

Configuration	
Authentication Type	<input checked="" type="radio"/> Form Based <input type="radio"/> Basic/NTLM
Start URI	
Form Method	POST
Form Action	http://weblogin.siterequest.com/sso/login.php
Form Parameter For User Name	user
Form Parameter For Password	pass
Hidden Form Parameters/Values	submit_form Submit \${session.logon.last.use

Figure 14.23 Hidden form parameters/values session variable for form-based AAA Server

Default customization parameters for Network Access

You can use session variable fields in the Network Access configuration, to specify the caption and detailed description for a network access resource that appears on the full webtop.

Example

In this example, the Caption and Detailed Description fields are populated with custom session variables, which are defined in the Variable Assign action for the access policy.

```
%(session.caption)
%(session.description)
```

Figure 14.24 Session variables for caption and detailed description fields

The screenshot shows a configuration window with two main sections: 'General Properties' and 'Customization Settings for English'.

**General Properties**

Name	na-docs
Partition / Path	Common
Description	

**Customization Settings for English**

Language	English
Caption	<code>%{session.caption}</code>
Detailed Description	<code>%{session.description}</code>
Image	<input type="text"/> <input type="button" value="Browse..."/> <a href="#">View/Hide</a>

**Figure 14.25** Caption and Detailed Description fields with custom session variables

## Network access launch application session variables

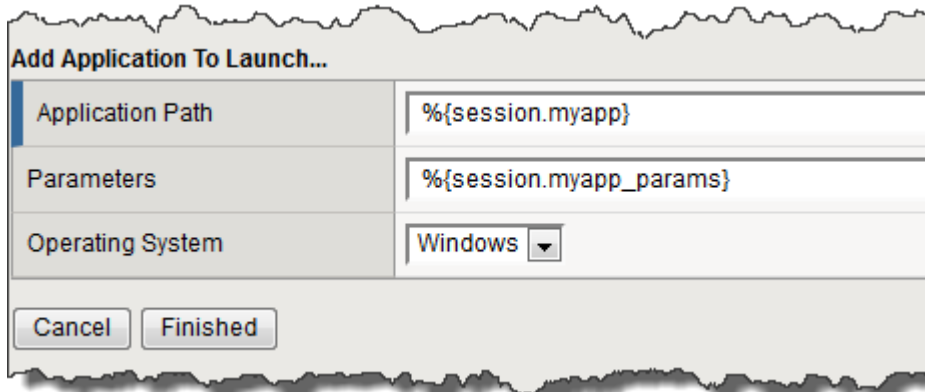
You can use session variable fields in the Network Access Launch Applications configuration, to specify the application path and application parameters for the auto launch applications for a network access resource.

### Example

In this example, the Application Path and Parameters fields are populated with custom session variables, that are defined in the Variable Assign action for the access policy.

```
%{session.myapp}
%{session.myapp_params}
```

**Figure 14.26** Session variables for network access application path and parameters fields



*Figure 14.27 Application path and parameters session variables in network access configuration*

## Network access drive mapping session variable

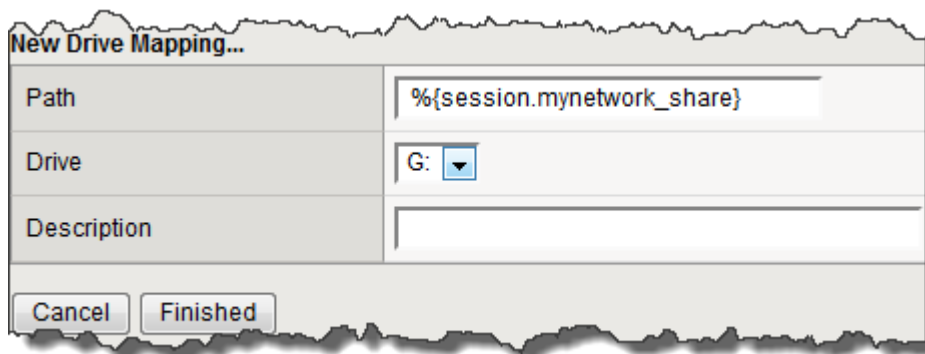
You can use a session variable in the Network Access Drive Mapping configuration, to specify the network path for the drive mapping for a network access resource.

### Example

In this example, the **Path** field is populated with the result from a custom session variable, which is defined in the Variable Assign action for the access policy.



*Figure 14.28 Session variable for network access drive mapping path field*



*Figure 14.29 Network access drive mapping path session variable*



## Application tunnel resource session variables

You can use session variables in the Application Tunnel configuration, to specify the destination, application path, and application parameters.

### Example

In this example, the **Destination** field, **Application Path**, and **Parameters** fields are populated with the result from custom session variables, which are defined in the Variable Assign action for the access policy.

```
%{session.myhost}
%{session.myapp}
%{session.myapp_param}
```

**Figure 14.30** Session variables for application tunnels

The screenshot displays the configuration interface for an application tunnel, organized into three main sections: General Properties, Compression Settings, and Launch Application.

- General Properties:**
  - Description:** ftp\_sessionvars
  - Destination:** Type: ☒ Host Name ☐ IP Address; Host Name: %{session.myhost}
  - Port(s):** Type: ☒ Port ☐ Port Range; Port: 21; FTP (dropdown)
  - Application Protocol:** FTP (dropdown)
  - Log:** None (dropdown)
- Compression Settings:**
  - Compression:** Disabled (dropdown)
- Launch Application:**
  - Application Path:** %{session.myapp}
  - Parameters:** %{session.myapp\_param}

At the bottom of the interface are two buttons: **Update** and **Delete**.

**Figure 14.31** Application tunnel session variables

## Remote desktop resource session variables

You can use session variables in the remote desktop configuration, to specify the destination, username source, password source, and domain source.

### Example

In this example, the Destination, Username Source, Password Source, and Domain Source fields are populated with the result from a custom session variable, which is defined in the Variable Assign action for the access policy, and the username, password, and domain are automatically populated with variables from the session.

```
%{session.mydesktop}  
session.logon.last.username  
session.logon.last.password  
session.logon.last.domain
```

*Figure 14.32 Session variables for remote desktops*

The image shows a configuration window titled "General Properties" for a resource named "rdp\_sessionvars". The window has a list of properties on the left and their corresponding values on the right. The properties and their values are:

Property	Value
Name	rdp_sessionvars
Type	Citrix
Description	
Destination	Type: <input checked="" type="radio"/> Host Name <input type="radio"/> IP Address Host Name: %{session.mydesktop}
Port	80
Server Side SSL	<input type="checkbox"/> Enable
ACL Order	After... /Common/test-foo
Log	None
Auto Logon	<input checked="" type="checkbox"/> Enable
Username Source	session.logon.last.username
Password Source	session.logon.last.password
Domain Source	session.logon.last.domain
Custom Parameters	

*Figure 14.33 Remote desktop resource session variables*





# 15

---

## Using Access iRule Events

---

- Introducing iRules
- Understanding ACCESS iRules
- Understanding ACCESS iRule Commands



## Introducing iRules

An **iRule** is a powerful and flexible feature within the BIG-IP® Local Traffic Manager™ system that you can use to manage your network traffic. Using syntax based on the industry-standard Tools Command Language (Tcl), the iRules® feature not only allows you to select pools based on header data, but also allows you to direct traffic by searching on any type of content data that you define. Thus, the iRules feature significantly enhances your ability to customize your content switching to suit your exact needs.

The remainder of this introduction presents an overview of iRules, lists the basic elements that make up an iRule, and shows some examples of how to use iRules to direct traffic to a specific destination such as a pool or a particular node.

### ◆ Important

*For complete and detailed information on iRules syntax, see the F5 Networks DevCentral web site at <http://devcentral.f5.com>. Note that iRules must conform to standard Tcl grammar rules; therefore, for more information on Tcl syntax, see <http://tmml.sourceforge.net/doc/tcl/index.html>.*

## What is an iRule?

An iRule is a script that you write. The iRules® you create can be simple or sophisticated, depending on your content-switching needs. Figure 15.1 shows an example of a simple iRule.

```
when CLIENT_ACCEPTED {
  if { [IP::addr [IP::client_addr] equals 10.10.10.10] } {
    pool my_pool
  }
}
```

**Figure 15.1** Example of an iRule

This iRule is triggered when a client-side connection has been accepted, causing the BIG-IP® system to send the packet to the pool **my\_pool**, if the client's address matches **10.10.10.10**.

Using a feature called the **Universal Inspection Engine**, you can write an iRule that searches either a header of a packet, or actual packet content, and then directs the packet based on the result of that search. iRules can also direct packets based on the result of a client authentication attempt.

iRules can direct traffic not only to specific pools, but also to individual pool members, including port numbers and URI paths, either to implement persistence or to meet specific load balancing requirements.

The syntax that you use to write iRules is based on the Tool Command Language (Tcl) programming standard. Thus, you can use many of the standard Tcl commands, plus a robust set of extensions that the BIG-IP system provides to help you further increase load balancing efficiency.

## Basic iRule elements

iRules® are made up of these basic elements:

- Event declarations
- Operators
- iRule commands

## Event declarations

iRules® are event-driven, which means that the BIG-IP® system triggers an iRule based on an event that you specify in the iRule. An ***event declaration*** is the specification of an event within an iRule that causes the BIG-IP system to trigger that iRule whenever that event occurs. Examples of event declarations that can trigger an iRule are **HTTP\_REQUEST**, which triggers an iRule whenever the system receives an HTTP request, and **CLIENT\_ACCEPTED**, which triggers an iRule when a client has established a connection.

Figure 15.2 shows an example of an event declaration within an iRule.

```
when HTTP_REQUEST {  
    if { [HTTP::uri] contains "aol" } {  
        pool aol_pool  
    } else {  
        pool all_pool  
    }  
}
```

**Figure 15.2** Example of an event declaration within an iRule



## Operators

An iRule operator compares two operands in an expression. In addition to using the Tcl standard operators, you can use the operators listed in Table 15.1.

Operator	Syntax
Relational operators	<code>contains</code> <code>matches</code> <code>equals</code> <code>starts_with</code> <code>ends_with</code> <code>matches_regex</code>
Logical operators	<code>not</code> <code>and</code> <code>or</code>

*Table 15.1 iRule operators*

For example, you can use the **contains** operator to compare a variable operand to a constant. You do this by creating an **if** statement that represents the following: "If the HTTP URI contains **aol**, send to pool **aol\_pool**." Figure 15.2, on page 15-2, shows an iRule that performs this action.

## iRule commands

An *iRule command* within an iRule causes the BIG-IP® system to take some action, such as querying for data, manipulating data, or specifying a traffic destination. The types of commands that you can include within iRules® are:

- ◆ **Statement commands**

These commands cause actions such as selecting a traffic destination or assigning a SNAT translation address. An example of a statement command is **pool <name>**, which directs traffic to the named load balancing pool.

- ◆ **Commands that query or manipulate data**

Some commands search for header and content data, while others perform data manipulation such as inserting headers into HTTP requests. An example of a query command is **IP::remote\_addr**, which searches for and returns the remote IP address of a connection. An example of a data manipulation command is **HTTP::header remove <name>**, which removes the last occurrence of the named header from a request or response.

- ◆ **Utility commands**

These commands are functions that are useful for parsing and manipulating content. An example of a utility command is **decode\_uri <string>**, which decodes the named string using HTTP URI encoding and returns the result.

## Understanding ACCESS iRules

This table includes session variables and related reference information for each session variable that you can use with Access Policy Manager®.

---

◆ **Note**

*iRule event access policy items must be processed and completed before the access policy can continue.*

### ACCESS\_SESSION\_STARTED

This event occurs when a new user session is created. This is triggered after creating the session context and initial session variables related to user's source IP, browser capabilities and accepted languages.

### Using ACCESS\_SESSION\_STARTED

This event provides a notification that a new session is created. You can use this event to prevent a session from being created when a specific event occurs. For example, if the user is exceeding the concurrent sessions limit, or if the user does not qualify for a new session due to custom logic, you can prevent a session from starting.

You can use ACCESS::session commands to get and set various session variables. Admin can also use TCP, SSL, and HTTP iRule commands to determine various TCP, SSL, or HTTP properties of the user.

### ACCESS\_SESSION\_STARTED examples

In this example, the system writes the browser user-agent to the log file when the session starts.

```
when ACCESS_SESSION_STARTED {  
    log local0.notice "APM: Received a new session from browser: [ACCESS::session data get  
    "session.user.agent"] "  
}
```

**Figure 15.3** ACCESS\_SESSION\_STARTED example logging browser user-agent

In this example, the system limits application access to the subnet **192.168.255.0** only.

```
when ACCESS_SESSION_STARTED {  
    set user_subnet [ACCESS::session data get "session.user.clientip"]  
    if { ($user_subnet & 0xffffffff) != "192.168.255.0" } {  
        log local0.notice "Unauthorized subnet"  
        ACCESS::session remove  
    }  
}
```

*Figure 15.4 ACCESS\_SESSION\_STARTED example limiting to a subnet*

## ACCESS\_POLICY\_COMPLETED

This event occurs when the access policy execution completes for a user session.

### Using ACCESS\_POLICY\_COMPLETED

This event provides a notification that access policy execution has completed for the user. You can use this event to perform post-access-policy work. For example, you can read and set session variables after the access policy is executed.

You can use `ACCESS::policy` and `ACCESS::session` commands to get and set various session variables. Admin can also use TCP, SSL, and HTTP iRule commands to determine various TCP, SSL, or HTTP properties of the user.

## ACCESS\_ACL\_ALLOWED

This event occurs when a resource request passes the access control criteria and is allowed through the ACCESS filter. This event is only triggered for resource requests and does not trigger for internal access control URIs such as **my.policy**.

### Using ACCESS\_ACL\_ALLOWED

This event notifies you that a resource request is being allowed to pass through the network. You can use this event to create custom logic that is not supported in a standard ACL.

For example, you can further limit access based on specific session variables, rate controls, or HTTP or SSL properties of the user.

You can use `ACCESS::session` commands to get and set session variables in this event, and `ACCESS::acl` commands to enforce additional ACLs.

## ACCESS\_ACL\_DENIED

This event occurs when a resource request fails to meet the access control criteria and is denied access.

### Using ACCESS\_ACL\_DENIED

This event provides notification that a resource request has been denied to pass through the network.

You can use this event to implement custom logic that is not supported in the standard ACLs. For example, you can send out a specific response, based on specific session variables, and HTTP or SSL properties of the user. This event may also be useful for logging purposes.

You can use `ACCESS::session` commands to get and set session variables in this event, and `ACCESS::acl` commands to enforce additional ACLs.

## ACCESS\_SESSION\_CLOSED

This event occurs when a user session is removed. This can occur because a user logs out, because the user session times out due to inactivity, or because the user session is terminated by an administrator.

You can use the `ACCESS::session` command to get session variables in this event. iRule commands which require a flow context can not be used in this event.

### Using ACCESS\_SESSION\_CLOSED

This event is used like `ACCESS_SESSION_STARTED`.

## ACCESS\_POLICY\_AGENT\_EVENT

This event allows you to insert an iRule event agent in an access policy at some point in the access policy:

On the server during access policy execution, the iRule event agent is executed and `ACCESS_POLICY_AGENT_EVENT` is raised in iRules®.

You can get the current agent ID (using an iRule command `ACCESS::policy agent_id`) to determine which iRule agent raised the event, and to do create some customized logic.

### Using ACCESS\_POLICY\_AGENT\_EVENT

Use this event to execute iRule logic inside TMM at the desired point in the access policy execution. For example, if you want to do concurrent session checks for a particular AD group, insert this agent after the AD query, and once user's group has been retrieved from AD query, check to see how many concurrent sessions exist for that user group in an iRule inside TMM.

## Understanding ACCESS iRule Commands

The following ACCESS iRule commands are available.

### ACCESS::disable

Use **ACCESS::disable** to disable access functionality for a specific request. When you disable access functionality, all access checks are skipped for the current request. This command is applied to a single request only. For the next request on the same connection or flow, the system will process access checks, unless another ACCESS::disable event is invoked.

This command allows you to bypass or disable access control features selectively for a backend application.

For example, if the backend server has URIs that you don't want to protect, or you want to allow access without a valid session, you can use ACCESS::disable in your iRule to disable access checking for those URIs.

Use this event with the HTTP\_REQUEST iRule event.

### ACCESS::session commands

The following commands are used with the ACCESS::session command.

#### ACCESS::session data get

This returns the value of session variable. Admin can read multiple session variables in the single instance of this command.

For example, **ACCESS::session data get "session.user.clientip"** gets the user's client IP address.

#### ACCESS::session data set

This sets the value of session variable to be the given . Admin can set multiple session variables in the single instance of this command.

For example, **ACCESS::session data set "myown\_custom\_variable" "my\_value"** creates the custom variable **myown\_custom\_variable**, and sets it to the value **my\_value**.

#### ACCESS::session remove

This deletes the user session and all associated session variables. The session is removed immediately after this command is invoked and no session variables can be accessed after this command.

ACCESS::session commands can be used only in ACCESS events.

## ACCESS::session exists

This command returns TRUE when the session with provided sid exists, and returns FALSE otherwise. This command is allowed to be executed in different events other than ACCESS events. One scenario for which you can use this command is to support a nonstandard HTTP application. The iRule verifies the MRHSession cookie, and provides a customized response that instructs the client to re-authenticate, as in the following example.

```
when HTTP_REQUEST {  
    set apm_cookie [HTTP::cookie value MRHSession]  
    if { $apm_cookie != "" && ! [ACCESS::session exists $apm_cookie] } {  
        HTTP::respond 401 WWW-Authenticate "Basic realm=\"www.example.com\""  
        return  
    }  
}
```

*Figure 15.5 ACCESS::session exists example*

## ACCESS::policy commands

The following ACCESS::policy commands are available.

### ACCESS::policy agent\_id

This returns the identifier for the agent raising the ACCESS\_CUSTOM\_EVENT.

### ACCESS::policy result

Returns the result of the access policy process. The result is one of the following:

- allow
- deny
- redirect

The ACCESS::policy command can only be used in ACCESS\_POLICY\_COMPLETED, ACCESS\_ACL\_ALLOWED and ACCESS\_ACL\_DENIED events.

### ACCESS::acl result

This returns the result of ACL match for a particular URI in ACCESS\_ACL\_ALLOWED and ACCESS\_ACL\_DENIED events.

This result can have one of the following values

- allow
- discard
- reject
- continue

## ACCESS::acl lookup

This returns the name of all the assigned ACLs for a particular session.

## ACCESS::acl eval \$acl\_name\_list

This applies all the acls specified in `acl_name_list` for a particular flow/URI.

ACCESS::acl commands can only be used in ACCESS\_ACL\_ALLOWED and ACCESS\_ACL\_DENIED events.

For example, to add an additional ACL named **additional\_acl** to a user's request before allowing it to go through, use the following example.

```
when ACCESS_ACL_ALLOWED {  
    ACCESS::acl eval "additional_acl"  
}
```

*Figure 15.6 ACCESS::acl eval example*







---

---

## Glossary

---

---



**absolute URL**

An absolute URL specifies the exact location of a file or directory on the internet.

**access control list (ACL)**

In Access Policy Manager®, the ACL is a set of restrictions associated with a resource or favorite that defines access for users and groups.

**access policy**

An access policy contains steps that the client and server go through before access is granted to a connection by the Access Policy Manager. See also *action*, *client side check*, *endpoint security*, *branch rule*.

**access profile**

An access profile is a pre-configured group of settings that you can use to configure secure network access for an application.

**action**

An action is an ordered set of rules for evaluating a remote system. Each action invokes one or more inspectors. The action then uses rules to test the inspectors' findings. In the visual policy editor, an action is depicted by a rectangle.

**Active Directory**

The Active Directory is a network structure supported by Windows 2000, or later, that provides support for tracking and locating any object on a network.

**advanced rules**

In an access policy, advanced rules provide customized functionality. This functionality is useful when you want more functionality than is provided by the default access policy rules and the rules created with the expression builder.

**allow ending**

An allow ending is a successful ending for the user in the access policy.

**authentication**

Authentication is the process of verifying the identity of a user logging on to a network.

**authentication action**

Authentication actions are used in an access policy to add an authentication check with a AAA server or with a client certificate.

**authentication query**

Authentication query searches the appropriate part of the directory tree structure of a AAA server, such as LDAP or Active Directory, to find a user within that directory.

**authorization**

Authorization is the process of enabling user access to resources, applications, and network shares.

**branch rule**

Branch rules test the inspectors' findings about a client system. The order of rules in a pre-logon sequence determines the flow of action.

**certificate**

A certificate is an online credential signed by a trusted certificate authority and used for SSL network traffic as a method of authentication.

**client certificate**

A client certificate enables the Access Policy Manager to verify the identity of a user's computer, and to control access to specific resources, applications, and files.

**client component**

A client component is a control downloaded from the Access Policy Manager that enables the various features of Access Policy Manager functionality.

**client side check**

In an access policy, a client side check defines a set of actions that need to be taken in order to evaluate the client system or device.

**Configuration utility**

The Configuration utility is the browser-based application that you use to configure the Access Policy Manager.

**decision box**

In the visual policy editor, a decision box is an policy action that provides a user with two options for accessing a system.

**domain name**

A domain name is a unique name that is associated with one or more IP addresses. Domain names are used in URLs to identify particular Web pages. For example, in the URL <http://www.siterequest.com/index.html>, the domain name is **siterequest.com**.

**Domain Name System (DNS)**

The Domain Name System (DNS) is a system that stores information associated with domain names, making it possible to convert IP addresses such as **192.168.16.8**, into more easily understood names such as **www.siterequest.com**.

**Dynamic Host Configuration Protocol (DHCP)**

DHCP is a protocol for assigning dynamic IP addresses to devices on a network. With dynamic addressing, a device can be assigned a different IP address every time it connects to the network.

**endpoint security**

Endpoint security is a centrally managed method of monitoring and maintaining client-system security. See also *client side check* and *resource protection*.

**FIPS**

Federal Information Processing Standards (FIPS) are publicly announced standards developed by the U.S. Federal government for use by all (non-military) government agencies and by government contractors. The Access Policy Manager can be configured with FIPS 140-encryption hardware, which stores all certificates and private keys in the FIPS hardware.

**FQDN (fully qualified domain name)**

The fully qualified domain name (FQDN) is an unambiguous domain name that specifies a node's position in the DNS tree hierarchy absolutely, for example, **myfirepass.siterequest.com**. See also *domain name*.

**high availability**

High availability is the process of ensuring access to resources despite any failures or loss of service in the setup. For hardware, high availability is ensured by the presence of a redundant system. See also *redundant system*.

**hot fix**

A hot fix (patch) is an intended modification to the BIG-IP® Access Policy Manager®.

**HTTP (HyperText Transport Protocol)**

HTTP is the method that is used to transfer information on the Internet and on intranets.

**HTTPS (HyperText Transport Protocol [Secure])**

HTTPS is secure HTTP. See also *HTTP (HyperText Transport Protocol)*.

**inspector**

An inspector is an ActiveX control or Java plug-in that gathers information about the user's computer, evaluating factors such as the presence of viruses or available software, operating system version, running processes, and others.

**interface**

A physical port on an F5 system is called an interface.

**IP address**

An IP address (Internet Protocol address) is a unique number that identifies a single device and enables it to use the Internet Protocol standard to communicate with another device on a network. See also *self IP address* and *virtual IP address*.

**IPsec**

IPsec (Internet Protocol Security) is a communications protocol that provides security for the network layer of the Internet without imposing requirements on applications running above it.

**local traffic management**

Local traffic management refers to the process of managing network traffic that comes into or goes out of a local area network (LAN), including an intranet.

**name resolution**

Name resolution is the process by which a name server matches a domain name request to an IP address, and sends the information to the client requesting the resolution.

**NAT (Network Address Translation)**

A NAT is an alias IP address that identifies a specific node managed by the Access Policy Manager system to the external network.

**network access**

Network access is a Access Policy Manager feature that provides secure access to corporate applications and data using a standard web browser.

**network configuration**

Network configuration is the process of setting up the Access Policy Manager's web services on network interfaces. See also *web service*.

**port**

A port is a number that is associated with a specific service supported by a host.

**redundant system**

Redundant system refers to a pair of units that are configured for failover. In a redundant system, there are two units, one running as the active unit and one running as the standby unit. If the active unit fails, the standby unit takes over and manages connection requests.

**resource**

A resource is an application, a file, or a server on your network to which you want users to have secure access.

**resource protection**

Resource protection is the process of using a defined protected configuration to protect a set of resources.

**self IP address**

A self IP address is an IP address that uniquely identifies each Access Policy Manager interface or VLAN interface. See also *IP address* and *virtual IP address*.

**sequence**

See *access policy*.

**server certificate**

A server certificate verifies the server's identity to a user's computer

**session variable**

A session variable contains a number or string that represents a specific piece of information about the client system, the Access Policy Manager, or another piece of information.

**split tunneling**

Split tunneling is a process that provides control over exactly what traffic is sent over the network access connection to the internal network.

**SSL (Secure Sockets Layer)**

SSL is a network communications protocol that uses public-key technology as a way to transmit data in a secure manner.

**standby controller/standby unit**

A standby unit in a redundant system is the unit that is always prepared to become the active unit if the active unit fails.

**strong password**

A strong password is one that is difficult to detect by both humans and computer programs, which effectively protects data from unauthorized access. A strong password typically consists of a specific number of alphanumeric characters of differing case, as well as certain punctuation characters.

**superuser**

Superusers are users who have cross-realm access to all groups and features. A superuser creates realm administrators, upgrading them from Access Policy Manager users, and delegating full or restricted access to Access Policy Manager functionality or groups.

**tunnel**

A tunnel is a secure connection between computers or networks over a public network.

**URI**

In the Access Policy Manager context, URI means the fully-qualified domain name, followed by the path designator */<uri-specific\_path>*.

**virtual host**

In the Access Policy Manager context, a virtual host means the domain name or IP address that users specify when logging on to a web service you create on a virtual IP. See also *virtual IP address*.

**virtual IP address**

A virtual IP address is an IP address that identifies a virtual (that is, non-physical) network location. The Access Policy Manager uses virtual IP addresses for redundant systems. See also *IP address*, *redundant system*, and *self IP address*.

**visual policy editor**

The visual policy editor consists of a graphical area in which you create, view, or modify an access policy by clicking to add and delete actions and rules that are visually shown on the graph. See also *access policy*, *action*, and *branch rule*.

**web service**

A web service is a method of communication that applications written in various programming languages and running on various platforms can use to exchange data over networks, such as the Internet or an intranet.

**webtop**

The webtop is the user's home page, which grants access to the network access connection.





---

---

# Index

---

---



/var/log/messages directory 12-5

32-bit registry keys  
     checking on 64-bit Windows 7-22  
 64-bit registry keys  
     checking 7-22

## A

access control  
     to SNMP data 13-3  
 access control entries  
     adding 3-3  
 access control list  
     assigning 3-5  
 access control lists  
     adding entries 3-3  
     and actions 3-3  
     and default actions 3-2  
     examples 3-6  
     logging 3-5  
     understanding 3-2  
 access levels, managing 13-5  
 access policy  
     adding a browser cache cleaner action 7-24  
     adding a client for MS Exchange check 8-15  
     adding a client OS check 8-2  
     adding a client type check 8-5  
     adding a decision box 6-18  
     adding a file check action 7-2  
     adding a landing URI check 8-11  
     adding a logon page 6-3  
     adding a machine cert check action 7-9  
     adding a macrocall 5-16  
     adding a message 6-18  
     adding a process check action 7-19  
     adding a protected workspace action 7-28, 7-37  
     adding a UI mode check 8-8  
     adding a virtual keyboard to the logon page 6-13  
     adding a Windows info action 7-11  
     adding actions 5-9  
     adding an external logon page 6-8  
     adding an IP geolocation match check 8-17  
     adding an iRule event 6-20  
     adding logging 6-17  
     and actions 4-2  
     and internal process for an action 4-8  
     and session variables 4-19  
     applying 5-4  
     assigning a dynamic ACL 3-12, 6-19  
     assigning a webtop 3-14  
     assigning an ACL 3-5  
     assigning resources 6-9

    assigning variables 6-10  
     configuring for systems that cannot use client-side checks 8-1  
     creating 5-7  
     filtering with Citrix Smart Access 6-15  
     logging session variables 6-17  
     selecting a route domain 6-16  
     setting a default ending 5-12  
     understanding basic configuration 5-8  
     understanding branches 4-12  
     understanding endings 5-10  
     understanding rules and actions 4-8  
 access policy ending  
     creating 5-10  
 Access Policy Manager  
     finding software version 1-27  
 access profile  
     creating 5-4  
     customizing languages 5-5  
     domain cookie option 5-3  
     export 5-21  
     import 5-21  
     secure cookie option 5-3  
     specifying a logout URI 5-2  
 ACL actions  
     allow 3-3  
     continue 3-3  
     discard 3-3  
     reject 3-3  
 actions  
     and internal process for 4-8  
     and pre-defined 4-3  
     and rules 4-8  
     that support MS Exchange clients 8-14  
     using in access policies 4-2  
 active connection statistics 13-15, 13-16  
 advanced access policy rules  
     and mcget command 11-17  
     creating a custom variable with 11-20  
     replacing configuration variable with custom expression 11-20  
     understanding situations 11-16  
     using 11-16  
     writing 11-17  
     writing in an action 11-18  
     writing in resource assign action 11-19  
 alarm RMON group 13-14  
 Alert log level 12-7  
 alert system 13-7  
 allow  
     in ACL 3-3  
 allow ending  
     configuring 5-11  
 allowed ending  
     understanding 4-17  
 an 3-5

- app tunnels
  - and SNAT 10-1
- application-specific MIB files 13-1
  - See also enterprise MIB files.
- apply access policy 5-4
- AskF5
  - and support 1-27
- assigning a dynamic ACL 3-12, 6-19
- assigning a webtop 3-14
- assigning an ACL 3-5
- assigning resources 6-9
- assigning variables 6-10
- audit log 12-2
- audit logging
  - and /var/log/ltm directory 12-5
  - enabling and disabling 12-8
- auditing events
  - and log levels 12-8
- authentication actions
  - understanding 4-7
- authentication warnings 13-8
- B**
- back up an access profile 5-21
- best practices
  - and client certificates 9-12
  - for certificate revocation lists 9-12
  - for Online Certificate Status Protocol 9-12
- BIG-IP alert system 13-2
- BIG-IP system information 13-3
- BIG-IP system objects, SNMP 13-2
- branch rules
  - and branches 4-12
  - examples 4-9
  - understanding 5-7
- branches in access policies 4-12
- browser cache cleaner action
  - understanding 7-24
  - using 7-24
- C**
- calculations 13-15
- certificate revocation list
  - and best practices 9-12
  - and limitation 9-11
  - described 9-11
- certificates
  - and Online Certificate Status Protocol 9-12
  - overview 9-2
  - understanding SSL server certificates 9-2
- Citrix resources
  - and SNAT 10-1
- Citrix Smart Access action
  - using 6-15
- client access
  - allowing 13-2, 13-4
  - configuring 13-3
- client certificates
  - and best practices 9-12
  - and certificate revocation list updates 9-12
  - and Online Certificate Status Protocol 9-12
- client for MS Exchange
  - supported actions 8-14
- client for MS Exchange check
  - using 8-15
- client OS check action
  - understanding 8-2
  - using 8-2
- client SSL
  - and DTLS hardware acceleration 10-4
- client type check
  - understanding 8-5
  - using 8-5
- clients, SNMP 13-3
- client-side checks
  - preparing for systems that cannot use 8-1
  - understanding 7-1
- collecting Windows information 7-11
- common operations, following recommended path 1-25
- communities
  - and access levels 13-5, 13-7
  - and trap destinations 13-8
- community access 13-5
- company-specific MIB files 13-1
- config variables
  - assigning 6-10
- configuration changes
  - auditing 12-5
- configuration data loads
  - logging 12-8
- configuration tasks
  - for SNMP agent 13-3
  - summary for SNMP 13-2
- Configuration utility
  - and components 1-20
  - and identification and messages area 1-20
  - and menu bar 1-20
  - and navigation pane 1-20
- configurations
  - and scenarios 1-26
- connection statistics 13-15, 13-17
- contact information 13-3
- contact name 13-3
- content searching 15-1
- content switching
  - customizing 15-1
- context-sensitive online help 1-27
- continue
  - in ACL 3-3
- Controlling SSL Traffic 9-1
- CPU use statistics 13-15, 13-19

Critical log level 12-7

CRL

See certificate revocation list.

## D

data

MIB files 13-13

data access control, SNMP 13-3

data loads

logging 12-8

data object values, SNMP 13-1

data objects

in MIB files 13-10

modifying 13-5, 13-7

See also access levels.

Debug log level 12-7

decision box action 6-18

default access control actions 3-2

default access levels

assigning 13-7

modifying 13-5

default ending 5-12

denied ending

understanding 4-17

deny ending

configuring 5-11

destinations, SNMP 13-7, 13-8

discard

in ACL 3-3

domain controller

IPv6 3-16

domain cookie option 5-3

domain mode

using with SSO 5-2

DTLS

configuring a virtual server 10-3

disabling hardware acceleration 10-4

dynamic access control list

creating 3-11

Dynamic ACL action

assigning a dynamic ACL 3-12, 6-19

## E

email, sending 12-1

Emergency log level 12-7

endings

and understanding 4-17, 5-10

creating 5-10

deny 4-17

for allowed users 4-17

for logon denied 5-10

for redirect 4-18

for webtop 5-10

redirect 5-10

setting default 5-12

endpoint security

and internal process for an action 4-8

and rule syntax 14-2

anti-spyware software check 7-40

antivirus check 7-40

firewall software check 7-40

hard disk encryption software check 7-40

patch management software check 7-40

peer-to-peer software check 7-40

understanding rules and actions in access policies 4-8

Windows health agent software check 7-40

enterprise MIB files

and Configuration utility 13-1

content of 13-11

defined 13-1

downloading 13-2, 13-10

Error log level 12-7

event notifications, SNMP 13-2

event RMON group 13-14

export an access profile 5-21

expr command

using 11-17

external logon page action

using 6-8

## F

F5 Technical Support, contacting 1-27

F5-BIGIP-COMMON-MIB.txt file 13-10

F5-BIGIP-LOCAL-MIB.txt file 13-10, 13-12

F5-BIGIP-SYSTEM-MIB.txt file 13-10, 13-13

fallback branch 4-12

file check action

understanding 7-2

using 7-2

## G

general purpose actions

configuring 6-1

global statistics data 13-13

graphs, SNMP 13-15

group policy

adding a template 7-36

downloading a template 7-36, 7-37

## H

header searching 15-1

help

locating online help 1-27

history RMON group 13-14

host names

in logs 12-2

HTTP request statistics 13-15, 13-18

**I**

- import an access profile 5-21
- information collection 13-2
- Information log level 12-7
- information polling 13-2
- information, SNMP 13-3
- interfaces
  - monitoring 13-14
- Introducing SSL server certificates 9-2
- IP address
  - with DTLS and network access virtual servers 10-3
- IP addresses
  - for SNMP traps 13-8
  - specifying 13-3
- IP geolocation match check
  - using 8-17
- iRule command types 15-3
- iRule elements 15-2
- iRule event declarations 15-2
- iRule operators 15-3
- iRules
  - defined 15-1
  - viewing reference 15-4
- irules
  - understanding 15-1

**L**

- landing URI check
  - using 8-11
- lease pools
  - creating 2-4, 2-5, 10-5
- local application traffic 13-12
- local traffic management information 13-10
- log contents 12-2
- log levels
  - defined 12-7
  - setting 12-7
- logging action
  - understanding 6-17
- logging session variables in an access policy 6-17
- logical operators 14-3
- logical operators, listed 15-3
- logon denied ending
  - customizing 5-13
  - understanding 5-10
- logon page
  - adding a virtual keyboard 6-13
  - customizing with logon page action 11-2
- logon page action
  - understanding 11-1
  - using 6-3
- Logout URI Include 5-2
- loopback interface 13-3

**M**

- machine cert check action
  - understanding 7-6
  - using 7-9
- machine location 13-3
- macro templates
  - for AD auth and resources 5-18
  - for SecurID and resources 5-19
- macro terminals
  - branches 4-12
  - configuring 5-15
  - understanding 4-15
- macrocalls
  - adding to an access policy 5-16
  - understanding 4-14
- macros
  - adding to an access policy 5-16
  - configuring 5-15
  - understanding 4-14
  - understanding terminals 4-15
- management information base
  - See also MIB-II MIB.
  - See MIB.
- mcget command
  - using 11-17
- memory use statistics 13-15, 13-16, 13-19
- message box action 6-17
- metrics collection 13-15
- MIB
  - and device management 13-1
  - defined 13-1
  - See also MIB-II MIB.
- MIB file contents 13-11
- MIB file locations 13-1
- MIB file types 13-10
- MIB files
  - defined 13-10
  - described 13-1
  - downloading 13-2
- MIB-II MIB 13-1
- MIB-II objects 13-13
- minimum log levels 12-1
  - defined 12-7
  - setting 12-7

- N**
- Net-SNMP 13-1
- network access
  - and SNAT 10-1
  - IP addresses and DTLS 10-3
- network access resource
  - assigning variable attributes 14-12
- network information 13-13
- new connection statistics 13-15, 13-17
- Notice log level 12-7

notification events 13-8  
 notification messages 13-2, 13-7  
     See also traps.  
 notifications, SNMP 13-12  
 NOTIFICATION-TYPE designation 13-12

## O

object data, SNMP 13-2  
 object ID definitions, RMON 13-14  
 object presentation 13-1  
 object values, SNMP 13-1  
 OIDs 13-15  
 Online Certificate Status Protocol  
     and best practices 9-12  
     using 9-12  
 online help 1-27  
 operating system-related events  
     logging 12-5  
 operators 15-3  
 optimized applications  
     and SNAT 10-1

## P

pager notifications, activating 12-1  
 partitions  
     and virtual servers 10-1  
 performance metrics, SNMP 13-2, 13-15  
 persistence  
     and iRules 15-1  
 platform information 13-13  
 policy-based routing 6-16, 11-10  
     example 11-12  
 port numbers 13-8  
 portal access  
     and SNAT 10-1  
 pre-defined actions 4-3  
 process check action  
     understanding 7-19  
     using 7-19  
 protected workspace  
     understanding 7-28  
 protected workspace action  
     using 7-28, 7-37

## Q

query commands, defined 15-3

## R

rate statistics 13-18, 13-21  
 read/write access level 13-5, 13-7  
 read-only access level 13-5, 13-7  
 redirect ending  
     configuring 5-11

understanding 4-18, 5-10  
 registry check action  
     and expression syntax for 7-21  
     specifying registry values 7-21  
     understanding 7-21  
 registry keys  
     checking on 64-bit Windows 7-22  
 reject  
     in ACL 3-3  
 relational operators, listed 15-3  
 release notes 1-27  
 remote desktops  
     and SNAT 10-1  
 Remote Network Monitoring  
     See RMON implementation  
 remote system management 13-3  
 Reporting 12-9  
 resource assign action  
     assigning a webtop 3-14  
     assigning an ACL 3-5  
     using 6-9  
 resources  
     and access control lists 3-2  
     domain controller 3-16  
     understanding 3-1  
 RFCDTLS cipher 10-4  
 RMON groups 13-14  
 RMON implementation 13-14  
 RMON-MIB.txt file 13-14  
 route domain selection action  
     using 6-16, 11-12  
 route domains  
     understanding 11-10  
 rule branches  
     adding actions 5-8  
 rule operators 14-3  
 rule operators, listed 15-3  
 rules  
     and actions in access policies 4-2  
     and session variables 4-19  
     and syntax elements 14-2  
     See iRules.  
     understanding 4-8  
     using 14-2  
     viewing predefined 4-10

## S

secure cookie option 5-3  
 security  
     and client-side checks 7-1  
 service names  
     in logs 12-2  
 session variables  
     and mcget command 11-17  
     assigning 6-10

- definition 4-20
- logging in an access policy 6-17
- understanding 4-19, 14-1
- using in access policies 11-16
- viewing reference 14-4

severity log levels

- defined 12-7
- setting 12-7

SNAT

- interactions 10-1
- understanding 10-1

SNAT information 13-10

SNMP

- and syslog 13-10
- configuring 13-7
- in the Configuration utility 13-4
- See also SNMP managers.
- See SNMP agent.

SNMP agent

- access to 13-4
- configuring 13-2
- defined 13-1

SNMP client 13-3

SNMP commands

- for collecting statistics 13-15
- using 13-2, 13-11

SNMP data access control 13-3

SNMP manager functions 13-2

SNMP managers

- as trap destinations 13-8
- defined 13-1

SNMP MIB files

- See MIB files.

SNMP object data 13-2

SNMP tasks 13-1, 13-2

SNMP traps

- handling 13-2
- See also traps.

SNMP user access 13-6

SNMP users 13-5

snmpd.conf files

- and access levels 13-6
- for trap configuration 13-8

snmpget command 13-15

software version, finding 1-27

SSL server certificates

- understanding 9-2

standard operators 14-2

statement commands

- defined 15-3

static access control list

- creating 3-3

statistical data 13-13

statistics

- and RMON group 13-14
- and SNMP 13-12

status codes

- in logs 12-2

successful branch 4-12

support

- and AskF5 1-27
- contacting F5 Networks Technical Support 1-27

system data, SNMP 13-12

system events

- logging 12-5

system information

- configuring 13-3
- polling for 13-12

system interface monitoring 13-14

system location 13-3

system messages

- viewing 1-20

system objects, SNMP 13-2

system-initiated changes

- logging 12-8

## T

task summary

- for SNMP 13-2

Tcl

- and logical operators 14-3
- and namespace sharing 11-18
- and rule operators 14-3
- and standard operators 14-2
- and validation 11-18
- using expr command 11-17
- using expression as a rule 11-17
- using mcget command 11-17
- using to write rules 11-17

Tcl syntax 15-2

Technical Support at F5, contacting 1-27

throughput rate statistics 13-15, 13-18, 13-21

timestamps

- in logs 12-2

Tools Command Language syntax 15-2

transaction IDs

- in logs 12-2

trap destinations

- configuring 13-3
- setting 13-7, 13-8

trap locations 13-10, 13-12

traps

- configuring 13-3
- defined 13-2, 13-7
- handling 13-2
- identifying 13-12

tree structure 13-1

two-factor authentication

- example 11-8



---

## U

UCD-SNMP 13-1

UI mode check  
    using 8-8

UIE commands, defined 15-3

UIE, defined 15-1

Universal Inspection Engine, defined 15-1

user changes

    logging 12-8

user names

    in logs 12-2

users

    See SNMP users.

    See user accounts.

using session variables 11-16, 14-11

## V

variable assign action

    understanding 6-10

    using 6-10

    using to replace configuration variable 11-20

version of software, finding 1-27

virtual keyboard action

    adding 6-13

virtual server

    defining for an access policy 10-2

virtual server information 13-10

virtual servers

    and DTLS 10-3

visual policy editor, starting 5-7

VLAN

    selecting in an access policy 11-10

VLAN gateway

    using with policy-based routing 11-10

VLAN selection action

    using 11-10

## W

Warning log level 12-7

warnings 13-8

webtop 3-14

    assigning 3-14

    creating 3-14

webtop ending

    understanding 5-10

Windows group policy

    adding templates 7-36

    downloading templates 7-36, 7-37

    understanding 7-32

Windows info action

    understanding 7-11

    using 7-11