F5® Networks
BIG-IP® Local Traffic Manager
Release: 10.2.2 Build 763.3 Hotfix 2 with
Advanced Client Authentication and Protocol Security Modules
Security Target
EAL 2 augmented ALC_FLR.2

Release Date:	March 8, 2013
Document ID:	09-2020-R-0062
Version:	D4.6
Prepared By:	Maryrita Steinhour F5 Networks
Prepared For:	F5 Networks 401 Elliott Avenue West Seattle, WA 98119

Table of Contents

List of Tables

List of Figures

Document History

Document Version	Date	Author	Comments
0.7	02/1/10	MMcAlister	Updates based on Verdicts issued 01/26/10 and via email informally
0.8	02/03/10	MMcAlister	Updated based on Verdicts issued 02/02/10
0.9	4/23/10	MMcAlister	Updated based on IVOR verdicts/response proposal from evaluators; corrected verdicts identified required for Project Kickoff (remainder to be addressed pre-TVOR)
D1.0	6/1/10	MMcAlister	Updated based on verdicts received via email
D1.1	6/4/10	MMcAlister	Updated based on informal verdicts
D1.2	6/10/10	MMcAlister	Added AOM to exclude list due to SSH, CLI exclusions
D1.3	04/14/11	MSteinhour	Update LTM version and build number, guidance document versions, SKUs Other minor clean-up updates
D1.4	05/26/11	MSteinhour	Minor updates noted in ALC evaluations
D1.5	06/08/11	MSteinhour	Clarify AOM
D1.6	06/14/11	MSteinhour	Clarify updates for appliance mode

D1.7	06/17/11	MSteinhour	Update per issues list
D1.8	07/12/11	RDay	Management clarifications
D1.9	07/18/11	MSteinhour, JCostlow	Clarifying per 7/14/11 verdicts draft
D2.0	07/25/11	MSteinhour	Additional updates on top of RDay's 7/22/11 updates
D2.1	07/29/11	MSteinhour	Additional updates on RDay's 7/27/11 updates
D2.2	09/13/11	MSteinhour	V1.8 verdict updates
D2.3	10/04/11	MSteinhour	ADV verdict updates
D2.4	11/09/11	MSteinhour	V1.9 verdict updates
D2.5	12/06/11	MSteinhour	V1.9 updated verdict responses
D2.6	12/22/11	MSteinhour	Exclude iControl
D2.7	01/09/12	MSteinhour	Sync with ADV_TDS updates
D2.8	01/10/12	MSteinhour	Remove FMT_REV.1
D2.9	02/15/12	MSteinhour	Editorial changes and changes to match ADV
D3.0	02/22/12	MSteinhour	Address comments on vD2.9
D3.1	02/28/12	MSteinhour	Address minor comments on D3.0
D3.2	04/11/12	MSteinhour	Address issues that surfaced during ADV_FSP review
D3.3	05/05/12	MSteinhour	Clarify traffic user / user role issues
D3.4	05/14/12	MSteinhour	Address minor comments from 5/11/12 review
D3.5	08/01/12	MSteinhour	Address comments/questions from TVOR
D3.6	08/06/12	MSteinhour	Address testing issues
D3.7	08/09/12	MSteinhour	Address additional testing issues
D3.8	08/30/12	MSteinhour	Address additional testing issues
D3.9	10/07/12	MSteinhour	Fix formatting, address additional issues
D4.0	10/20/12	MSteinhour	Clarifications and logging details
D4.1	10/30/12	MSteinhour	Updates per 2012-10-29 comments
D4.2	11/19/12	MSteinhour	Minor updates
D4.3	11/30/12	MSteinhour	Update sorting capability for FAU_SAR.3

D4.4	12/12/12	MSteinhour	Final updates
D4.5	02/08/13	MSteinhour	FVOR updates
D4.6	03/08/13	MSteinhour	FVOR cleanup

1 Introduction

This section identifies the Security Target (ST), Target of Evaluation (TOE), conformance claims, ST organization and terminology. It also includes an overview of the evaluated product.

1.1 Identification

TOE Identification: F5 Networks BIG-IP® Local Traffic Manager Release 10.2.2 Build 763.3 with Hotfix-BIGIP-10.2.2-911.0-HF2 plus the Advanced Client Authentication and Protocol Security Modules and Appliance Mode License running on Model 11050, 8900, or 6900 redundant pair hardware platform (quantity 2).[1]
See section 1.6.1.1 for the list of specific SKUs/PNs included in the TOE.

ST Identification: F5® Networks BIG-IP® Local Traffic Manager Release: 10.2.2 Build 763.3 with Hotfix 2 with Advanced Client Authentication and Protocol Security Modules and Appliance Mode License Security Target EAL 2 augmented ALC_FLR.2

ST Version: D4.6

ST Publish Date: March 8, 2013

ST Author: Maryrita Steinhour, F5 Networks

1.2 Organization

- **Security Target Introduction (Section 1)** – Provides identification of the TOE and ST, an overview of the TOE, an overview of the content of the ST, and relevant terminology. The introduction also provides a description of the TOE security functions as well as the physical and logical boundaries for the TOE, the hardware and software that make up the TOE, and the physical and logical boundaries of the TOE.
- **Conformance Claims (Section 2)** – Provides applicable Common Criteria (CC) conformance claims, Protection Profile (PP) conformance claims, and Assurance Package conformance claims.
- **Security Problem Definition (Section 3)** – Describes the threats, organizational security policies, and assumptions pertaining to the TOE and the TOE environment.
- **Security Objectives (Section 4)** – Identifies the security objectives for the TOE and its supporting environment as well as a rationale that objectives are sufficient to counter the threats identified for the TOE.
- **Extended Components Definition (Section 5)** – Presents components needed for the ST but not present in Part II or Part III of the Common Criteria Standard.

[1] The redundant pair is required to be two (2) TOE hardware appliances of the same model.

- **Security Requirements (Section 6)** – Presents the Security Functional Requirements (SFRs) met by the TOE and the security functional requirements rationale. In addition this section presents the Security Assurance Requirements (SARs) met by the TOE as well as the assurance requirements rationale. Provides pointers to all other rationale sections, to include the rationale for the selection of IT security objectives, requirements, and the TOE summary specifications as to their consistency, completeness, and suitability
- **Summary Specification (Section 7)** – Describes the security functions provided by the TOE that satisfy the security functional requirements, provides the rationale for the security functions. It also describes the security assurance measures for the TOE as well as the rationales for the assurance measures.

1.3 Document Terminology

Please refer to CC v3.1 Part 1 Section 4 for definitions of commonly used CC terms.

1.3.1 ST Specific Terminology

Address Resolution Protocol	A network protocol, which maps a network layer protocol address to a data link layer hardware address.
Administrative-user(s)	Refers to a user of the BIG-IP appliance (not of the traffic it mediates) holding any of the supported roles as described in section 6.2.4.3; used to globally characterize TOE users within this ST.
Administrative-user, Default	"Default Administrative-user" refers specifically to the one administrative-user required to be locally authenticated and available in case remote access is not possible.
Administrative Partition	An Administrative Partition is a logical container that Administrators can create, containing a defined set of BIG-IP system objects.
Backend Servers	Within this ST, this term refers to the group of servers (aka pool members), organized in Pools, which are served by the BIG-IP appliance.
Broadcast network	A network where traffic is sent (broadcast) to all devices on the network, rather than routing to a specific device.
Client-side Traffic	Refers to connections between a client system and the BIG-IP appliance.
Content Server	Within this ST, a content server refers to the BIG-IP supported web or application (client) servers. Content servers are a subset of backend servers.
Cyclic Redundancy Check	A non-secure checksum designed to detect accidental changes to raw computer data.
Certificate Revocation List (CRL)	Refers to a listing of certificates that have been revoked by an applicable authority. An authenticating system checks a

	CRL to see if the SSL certificate that the requesting system presents for authentication has been revoked.
Data Guard™	An F5 Networks feature where data matching configured patterns is anonomized with characters to obscure sensitive data in transit.
External addresses	IP addresses on the external network
External network	Within this ST, the network between the BIG-IP appliance and its client systems.
Internal addresses	IP addreses on the internal network
Internal Network	Within this ST, the network between the BIG-IP appliance and the backend servers.
iRules™	iRules refers to an F5 Networks scripting language included in the BIG-IP that may be used by users to develop scripts that control the behavior of a connection passing through the BIG-IP appliance.
iRules script	A script created using the iRules scripting language.
Local traffic management	The process of managing network traffic that comes into or goes out of a local area network (LAN), including an intranet.
Loopback network	In IPv4, the network with the CIDR prefix 127/8; ::1/128 in IPv6.
Node	An IP Address configured within BIG-IP as a destination.
OneConnect™	The F5 Networks OneConnect feature optimizes the use of network connections by keeping server-side connections open and pooling them for re-use.
Persistence	When load balancing, persistence means that once a client is connected to a specific server, future connections are always sent back to the specific server.
Persistence cookies	A method of using HTTP cookies to make connections persistent.
Pool	A grouping of backend servers.
Pool Member	This term refers to node-and-service pairs which are assigned to one or more Pools.
Protocol Security Module	The BIG-IP Protocol Security Module (PSM) runs on the BIG-IP traffic management platform, providing application security functionality.

Protocol Aware	This term refers to the fact that the TMM can readily identify protocols that flow on top of TCP, such as HTTP SMTP and protocols that flow under TCP such as routing protocols. Since TMM's functionality includes decoding these protocols, extra information about the traffic stream can be extracted and applied for firewall functionality.
Protocol Sanitization	Refers to Protocol RFC based compliance checks performed by BIG-IP. For the CC Evaluated configuration these are HTTP, FTP, and SMTP.
Server-side traffic	Refers to connections between the BIG-IP appliance and a target server system (backend server).
SSL	Secure Socket Layer. When used without a version number, this refers to the SSLv3/TLSv1.0 protocol support. Note that Profiles and iRules both refer to "SSL"; the support includes SSLv3/TLSv1.0 protocols even when the actual GUI page or command string is "SSL".
TLS	Transport Layer Security. When used without a version number, this refers to the TLSv1.0 protocol support.
Traffic, Administrative	Traffic generated in order to manage the TOE (e.g. traffic from the administrative user to the GUI or tmsh).
Traffic user	(also client/server traffic users) Users sending traffic through the TOE; so named as to distinguish them from Administrative-users.
Traffic, User-generated	All traffic other than administrative traffic that flows through the TOE.
Virtual address	A virtual address is an IP address associated with one or more virtual servers managed by the BIG-IP system.
Virtual port	A virtual port is the port number or service name associated with one or more virtual servers managed by the BIG-IP system. A virtual port number should be the same TCP or UDP port number to which client programs expect to connect.
Virtual server	Virtual servers are a specific combination of virtual address and virtual ports, associated with a content site that is managed by a BIG-IP system or other type of host server. Also includes VLAN and protocol (TCP vs. UDP).

VLAN (virtual local area network)	A VLAN is a logical grouping of interfaces connected to network devices. A VLAN may be used to logically group devices that are on different network segments. Devices within a VLAN use Layer 2 networking to communicate and define a broadcast domain.

1.3.2 Acronyms

ACA	Advanced Client Authentication (module)
ARP	Address Resolution Protocol
ASM	Application Security Module
CRL	Certificate Revocation List
CC	Common Criteria
CRC	Cyclic Redundancy Check
DoS	Denial of Service
FTP	File Transfer Protocol
GTM	Global Traffic Management
GUI	Graphical User Interface
HTTP	Hypertext Transport Protocol
HTTPS	Hypertext Transport Protocol (Secure)
LTM	Local Traffic Management
OCSP	Online Certificate Status Protocol
OS	Operating System
OSI	Open Systems Interconnection
PAM	Pluggable Authentication Module
PSM	Protocol Security Module
SNAT	Secure Network Address Translation
SSL	Secure Socket Layer
SMTP	Simple Mail Transfer Protocol
SFP	Security Function Policy
SPF	Sender Policy Framework
TCP	Transmission Control Protocol
TLS	Transport Layer Security

TMOS (also TM/OS)	Traffic Management Operating System
TOE	Target of Evaluation
URI	Uniform Resource Indicator
VLANs	Virtual Local Area Networks
VNIC	Virtual Network Interface Card (driver)

1.4 Common Criteria Product type

The TOE is classified as a **Switch/Router** for Common Criteria purposes. The TOE is made up of hardware and software components. The TOE consists of an appliance that is deployed in a quantity two (2), "redundant pair" configuration.

1.5 Overview

The BIG-IP device is a port-based, multilayer switch with multiple ports and a powerful host system for advanced processing. The system reduces the need for routers and IP routing by managing traffic at the data-link layer (Layer 2). The multilayer capability of the BIG-IP system provides the ability for the system to process traffic at OSI layers 2 and above. The BIG-IP system performs basic Layer 4 load balancing and is fully capable of managing traffic at Layer 7. The system performs IP routing at Layer 3 when needed, and manages TCP and application traffic at Layers 4 and 7. The BIG-IP (TOE) also includes the Application Client Authentication and Protocol Security Modules which are included in appliance software and are enabled through licensing for the CC Evaluated configuration.

The BIG-IP system provides the ability to monitor the devices for which it manages traffic and to provide audit trails relating to the use of network resources. BIG-IP information flow control rules ensure that critical connections using IP protocols reach the correct destination server. The BIG-IP appliance supports HTTP, SMTP, and FTP routing and analysis and can be configured to perform analysis on all other Ethernet/IP based protocols using the iRules scripting feature. Using packet filtering and profile based routing provided by the Protocol Security Module (PSM), the TOE protects backend servers from unsolicited traffic and potentially malicious traffic flows. The PSM also performs security related checks and validations for HTTP, SMTP, and FTP traffic. Key Traffic Management features provided by BIG-IP include:

- Processing of SSL session authentication and SSL encryption to improve server performance.
- Client/Server Certificate based authentication of SSL traffic provided through the installed Advanced Client Authentication Module (ACA).
- Establishing and managing session and connection persistence.
- Handling application-traffic authentication and authorization functions based on User name/password and SSL certificate credentials.
- Protocol Sanitization – Terminates all TCP connections, preventing out-of-order packet floods, MSS tiny packet floods and TCP window tampering. Includes

HTTP header evaluation, RFC violation matches, and protocol enforcement checks provided through the installed Protocol Security Module.

- Customizing the flow of application-specific traffic (such as HTTP and SSL traffic).
- Customizing the management of specific connections according to user-written scripts using iRules. iRules is based on the industry-standard Tool Command Language (TCL).

Through the use of the proprietary scripting language, iRules™, traffic can be routed based on a rules driven configuration scripts to optimize traffic flows based on pre-configured conditions. iRules™ can be used to produce a user-written script that controls the behavior of a connection passing through the system.

The BIG-IP system also enhances network security through features such as Denial of Service (DoS) protection and application filters. DoS features include protocol compliance based checks, management of TCP requests using maximum thresholds, throttling of traffic based on memory usage, and use of SYN cookies that prevent SYN ACK based resource exhaustion. In addition to these features, the Network Administrator can configure the BIG-IP appliance to offload processor intensive SSL processing from backend servers and secure traffic destined to the server pools based on a variety of encryption algorithms. In this role, the TOE can provide authentication services for traffic flows on behalf of backend servers assuring that unauthenticated traffic is not permitted to access server resources. Through the Advanced Client Authentication (ACA) module, BIG-IP provides certificate based authentication of SSL traffic and certificate revocation through OCSP.

The BIG-IP appliance, Common Criteria Evaluated configuration provides analysis functions, as described herein, for the following protocols:

- HTTP
- SMTP
- FTP

Though the use of iRules scripts, the BIG-IP appliance can be configured to analyze and process any Ethernet/IP based protocol. Non-Ethernet protocols cannot be analyzed by the BIG-IP appliance.

The BIG-IP® system also provides non-security related performance enhancements which are not evaluated as part of the Common Criteria Evaluation. These include the ability to load balance and optimize network and application traffic by using compression, caching data, using session persistence, and other traffic optimization techniques. In addition, monitors provide the ability to route connections around slower or degraded resources, and as a result, critical connections are made using the optimum route. The output of the monitors provides the ability for the Network Administrator to view network efficiency.

1.6 TOE Description

1.6.1 Physical Boundaries

The Physical boundary of the TOE is the BIG-IP appliance and installed software. The TOE is delivered with software pre-installed on the appliance hard drive resource.

1.6.1.1 Hardware

BIG-IP hardware appliance: Hardware Chassis Model 11050, 8900, or 6900 hardware platform (quantity 2)

a. Model: 6900
SKU: F5-BIG-LTM-6900-8G-R
PN: 200-0300-01

or

b. Model: 8900
SKU: F5-BIG-LTM-8900-R
PN: 200-0308-01

or

c. Model: 11050
SKU: F5-BIG-LTM-11050-R
PN: 200-0299-00

Three hardware platforms are available for the CC Evaluated configuration of the BIG-IP appliance, the 11050, 8900, and 6900 model chassis. All three hardware options share the same software build installation and are differentiated by throughput. The model arrangement includes progressively more processors and RAM to accommodate greater throughput. The model 11050 platform includes dual, hex core processors with 32 GB RAM, the 8900 platform deploys 2 quad core processors with 16 GB RAM, and the 6900 platform features 2 dual core processors with 8 GB RAM. All platforms include two 320 GB hard drives. The 8900 and 6900 models provide 16 gigabit Ethernet copper ports and 8 gigabit fiber ports. The 11050 model provides 10x 10G SFP+ ports (which can support LX, SX, Copper, and SR connectors) and includes 2x fiber transceivers. The 8900 chassis can be fitted with up to two 10-gigabit fiber ports which may be considered as a non-security-relevant part of the TOE.

Specifications	11000 Series (Model 11050)	8900 Series (Model 8900)	6900 Series (Model 6900)
Traffic Throughput	40 Gbps	12 Gbps	6Gbps
Hardware SSL	Included: 500 TPS Maximum: 116,000 TPS, 19.2 Gbps Bulk encryption	Included: 500 TPS Maximum: 58,000 TPS, 9.6 Gbps Bulk encryption	Included: 500 TPS Maximum: 25,000 TPS, 4 Gbps Bulk encryption
FIPS SSL	FIPS 140-2 Level 2 (Option) 20,000 TPS	FIPS 140-2 Level 2 (Option) 20,000 TPS	FIPS 140-2 Level 2 (Option) 20,000 TPS
Hardware Compression		Included: 50 Mbps Maximum: 8 Gbps	Included: 50 Mbps Maximum: 5 Gbps
Software Compression	Included: 50 Mbps Maximum: 12 Gbps		
Processor	Dual CPU, Hex Core (12 processors)	Dual CPU, Quad Core (8 processors)	Dual CPU, Dual Core (4 processors)
Memory	32 GB	16 GB	8 GB
Hard Drive	Two 320 GB drives	Two 320 GB drives	Two 320 GB drives
Gigabit Ethernet CU Ports		16	16
Gigabit Fiber Ports (SFP)		8 LX; SX or Copper (4 SX included)	8 LX; SX or Copper (4 SX included)
10 Gigabit Fiber Ports (SFP+)	10 SR (2 included)	2 SR (Sold Separately)	
Power Supply	Dual 850W included	Dual 850W included	Dual 850W included
Typical Consumption	397 W (110V input)	450W (110V input)	300W (110V input)
Input Voltage	90-240 VAC +/- 10% auto switching, 50/60 hz	90-246 VAC +/- 10% auto switching, 50/60 hz	90-246 VAC +/- 10% auto switching, 50/60 hz
Typical Heat Output	1536 BTU/hour (110v input)	1536 BTU/hour (110v input)	1024 BTU/hour (110v input)
Dimensions	5.2"H x 17.4"W x 21.4"D 3U industry standard rack-mount chassis	3.5"H x 17.3"W x 21.4"D 2U industry standard rack-mount chassis	3.5"H x 17.3"W x 21.4"D 2U industry standard rack-mount chassis
Weight	52.0 lbs. (dual power supply)	45.5 lbs. (dual power supply)	45.5 lbs. (dual power supply)

Figure 1: BIG-IP Appliance: Hardware specifications

*Note: "FIPS SSL" options listed above are not included in the CC Evaluated Configuration.

1.6.1.2 Software

Software for the BIG-IP platform consists of the following F5 components:

a. BIG-IP® Local Traffic Manager Release 10.2.2 Build 763.3 with Hotfix-BIGIP-10.2.2-911.0-HF2.

b. Protocol Security Module (PSM) (F5-ADD-BIG-PSM)

c. Advanced Client Authentication (ACA) module (F5-ADD-BIG-ACA)

d. BIG-IP ADD-ON: Appliance Mode License (restricts the CLI to tmsh only; no bash access and no ability to login as root) (F5-ADD-BIG-MODE)

Software for BIG-IP Platform includes the following third party components:

- CentOS 5.4, Linux kernel 2.6
- openssl 0.9.8e (non-FIPS validated version)
- openssh 4.3p2 (the TOE support is restricted to SSHv2)
- Apache Httpd 2.2.3
- Tomcat 6.0.20
- Java 1.6.0

1.6.2 Guidance Documents

The following guidance documents are provided for download through the F5 Networks support website and/or delivered with the product in accordance with EAL 2 requirements:

PRODUCT DOCUMENTATION CONFIGURATION ITEMS			
	Description of CI	**Document Name**	**Document Number and Revision**
Softcopy Documents			
1	Platform Guide	Platform Guide: 6900	MAN-0329-00
2	Platform Guide	Platform Guide: 8900	MAN-0330-00
3	Platform Guide	Platform Guide: 11050	MAN-0322-01
4	TMOS Management Guide	TMOS™ Management Guide for BIG-IP® Systems version 10.1	MAN-0294-01
5	Configuration Guide - LTM	Configuration Guide for BIG-IP® Local Traffic Manager™ Version 10.1	MAN-0292-01
6	Getting Started Guide	BIG-IP® Systems: Getting Started Guide version 10.1	MAN-0300-00
7	Configuration Guide - PSM	Configuration Guide for BIG-IP® Protocol Security Module	MAN-0284-02

		version 10.2	
8	Tmsh Guide	Traffic Management Shell (tmsh) Reference Guide version 10.2	MAN-0306-01
9	Installing hotfixes	SOL10025: Managing F5 product hotfixes for BIG-IP version 10.x systems	SOL10025 Updated 08/19/2011
10	Common Criteria Guidance Wrapper	Common Criteria Guidance Supplement	10-2020-R-0039 v3.0
11	iRules	iRules-10.2.zip	No document number
12	iRules Details	iRules Details-10.2.2.zip	No document number
13	Security Vulnerability Response Policy	SOL4602 – Overview of the F5 security vulnerability response policy	SOL4602 Updated 08/16/2012
Shipped in hardcopy form with the appliance and available in softcopy – all platforms			
14	F5 Networks Terms of License and Sale	F5 Networks Terms of License and Sale	PUB-0024-04 Rev. B
15	End User Software License	End User Software License	PUB-0026-09 (not listed on document) 2011-05-16
16	Support Contact flyer	Support Contact flyer	PUB-0093-03
17	Configuration Worksheet	Configuration Worksheet F5 BIG-IP Local Traffic Manager	PUB-0090-02
18	EU Battery Notice	European Union Battery Notice	PUB-0186-01 Rev A
19	Quick Start Flyer	Quick Start Flyer	PUB-0228-00 (not listed on document)
20	Letter to Customer	Letter to Customer	No document number
Shipped in hardcopy form with the appliance and available in softcopy – 6900 platform			

21	EC Declaration of Conformity	EC Declaration of Conformity (6900 and 8900 platforms)	PUB-0209-02 Rev A
22	Setting up the 6900/8900/8950 Platform	Setting up the 6900/8900/8950 Platform	MAN-0288-02
23	6900/8900/8950 Packing List	6900/8900/8950 Platform Packing List	PUB-0201-02 Rev A
24	6900/8900/8950 Hazardous Substance Table	6900/8900/8950 Platform Hazardous Substance Table	DOC-0300-01
Shipped in hardcopy form with the appliance and available in softcopy – 8900 platform			
25	EC Declaration of Conformity	EC Declaration of Conformity (6900 and 8900 platforms)	PUB-0209-02 Rev A
26	Setting up the 6900/8900/8950 Platform	Setting up the 6900/8900/8950 Platform	MAN-0288-02
27	6900/8900/8950 Packing List	6900/8900/8950 Platform Packing List	PUB-0201-02 Rev A
28	6900/8900/8950 Hazardous Substance Table	6900/8900/8950 Platform Hazardous Substance Table	DOC-0300-01
Shipped in hardcopy form with the appliance and available in softcopy – 11050 platform			
29	EC Declaration of Conformity	EC Declaration of Conformity (11050 platforms)	PUB-0223-02 Rev A
30	Setting Up 11000 Series Platforms	Setting Up 11000 Series Platforms	MAN-0323-02
31	11050 Packing List	11050 Platform Packing List	PUB-0200-02 Rev A
32	11050 Hazardous Substance Table	11050 Platform Hazardous Substance Table	DOC-0306-00

Table 1 Product Documentation Configuration Items

1.7 Operational Environment Resources

The Operational Environment resources required by the TOE are defined in sections 1.7.1, 1.7.2, and 1.7.3:

1.7.1 General Resources

- Network Time Protocol (NTP v4.2.2p1 or later) server
- OCSP (v1 or later) server supporting certificate revocation checking
- General purpose commodity gigabit Ethernet switches or a general purpose VLAN capable switch

1.7.2 Hardware

The following table identifies hardware components for the Operational Environment:

Boundary	Component	Description
Environment	Console Workstation	Console Platform supporting either the browser interface used for GUI sessions or the ssh client for tmsh sessions

Table 2: Operational Environment Hardware Components

1.7.3 Software

The following table identifies software components for the Operational Environment:

Boundary	Component	Description
Environment	• Microsoft® Internet Explorer®, version 7.0x or later • Mozilla® Firefox®, version 3.0x or later	Browser installed on the Administrator console machine for establishing Console sessions; must support SSL sessions using AES 128 or 256. If you plan to use the BIG-IP Dashboard (Overview -> Dashboard from the GUI Main page) to view statistics, the browser must have Abobe™ Flash Player version 9 or later installed. Note: The TOE was tested with Microsoft® Internet Explorer®, version 9.0 and Mozilla® Firefox®, version 10.0.
Environment	• SSH client	PuTTY v0.62 or equivalent SSH client.

Table 3: Operational Environment Software Components

1.7.4 Recommended Operational Enivoronment resources

The following Operational Environment resources are strongly recommended but not required.

- Mail Server (supporting SMTP (RFC 2821)) for security violation alerts
- Backend Content Server resources (supporting HTTP v1.1 with SSLv3 and TLSv1, FTP (RFC 959), SMTP (RFC2821))
- Syslog server (v2.0.8 or later) for log offloading from the BIG-IP appliance
- External authentication server (LDAP (v3 or later) or RADIUS (RFC 2865))

1.8 Logical Boundaries

This section describes the product features and denotes which are within the logical boundaries of the TOE. The following Security Functions are supported by the F5 Networks BIG-IP appliance:

- Security Audit
- Identification and Authentication
- Security Management
- Secure Communications
- Secure Traffic
- Protection of the TSF
- User Data Protection: Information Flow Control
- Email alerts (if configured)

1.8.1 Security Audit

The TOE generates 5 types of logs as part of the security audit security function: System Events, Packet Filter Events, Local Traffic Events, Audit, and Application Security (PSM) logs. Within this ST, these categories of audit logs are collectively referred to as Audit logs or Audit records, unless otherwise stated.

System Events logs refer to logging of an event associated with the underlying Linux based Operating System. Packet Filter Event logs pertain to event logs generated through Packet Filter rules in effect during operation. Local Traffic Event logs relate to event logs generated through Local Traffic Management functions of the TOE (such as routing). Audit Event logs pertain to the logging of configuration changes and other settings made on the appliance by administrative-users.

Local Traffic and Audit (type) logs may be configured to various logging levels based on the type of events will trigger the generation of a log record. By default, the Common Criteria Evaluated configuration enables all logs and specifies the audit level necessary to fully comply with audit requirements. Instructions relating to the configuration of the BIG-IP appliance audit

logging levels to meet these requirements are specified in Common Criteria Administrator Guidance.

Application security logs record security violation events that are triggered based on traffic that violates security profile rule criteria established by default behavior and authorized administrative-user configured parameters. These security violations are managed by the PSM module of the TMM and are described below under Section 1.8.7: User Data Protection: Information Flow Control.

Audit records are stored local to the appliance and may be exported to a Syslog server in the Operational Environment.

Audit records within the TOE may be selectively filtered and searched based on various characteristics. Audit records are accessed through the Administrator Console GUI or tmsh. Audit records cannot be modified by any user. Protection of the audit system is provided by the underlying BIG-IP TMOS and TOE identification and authentication mechanisms.

1.8.2 Identification and Authentication

The BIG-IP TOE allows access to security management functions only to those administrative-users who are identified and authenticated as holding a valid account.

Both local (within the appliance) and remote identification and authentication mechanisms are available. The default Administrative-user is required to be locally authenticated to assure access to the appliance is possible during a loss of connectivity to the external authentication server. Administrative-users of the TOE appliance are exclusively authenticated using a username / password combination.

The BIG-IP manages identification and authentication for Administrative-user access to the TOE through the PAM (Pluggable Authentication Modules) module installed as part of the underlying Linux based operating system and configured via server profile configuration. One PAM handles the local authentication while others handle the interfaces to the remote authentication servers.

The types of remote authentication servers that can be used to store user accounts for BIG-IP TOE Administrative-users are:

- Lightweight Directory Access Protocol (LDAP) servers
- Remote Authentication Dial-in User Service (RADIUS)

The Common Criteria Evaluated Configuration recommends the use of an external authentication server in the Operational Environment (LDAP or RADIUS) for authenticating all users except the default Administrative-user.

Authenticated traffic flow is discussed below in Section 1.8.7, Information Flow Control. This usage of "authentication" relates to data protection and not the Identification and Authentication security function. SSL traffic may be authenticated through X.509 certificates as configured through configured authentication profiles within the appliance. Validation checks may be performed internal to the appliance or externally. Revocation checks are performed externally through the use of an OCSP server in the Operational Environment.

Cookies passed to clients as part of session initiation may be encrypted to assure that they are unmodified and being used by the same entity that initiated the session.

1.8.3 Security Management

Security Management is managed by authorized Administrative-users utilizing the BIG-IP TMOS through the Administrator Console GUI. The TOE provides the following roles for administering the appliance:

- Administrator – Full Access user
- Resource Administrator – Object based access except user account data
- User Manager – Access to user account data except Administrator level
- Manager – Access to create, manage, and delete traffic management objects such as virtual servers/pools
- Application Editor – Access to modify traffic management objects such as virtual servers/pools
- Operator – Access to enable/disable nodes & pool members
- Guest – Read access to all object, may change own passwords
- Web Application Security Administrator, Web Application Security Editor – Read access to all objects, may change own password, access to modify Application Security Module objects.
- No Access – No access to any objects; initiate state when creating new user

In addition to the assigned roles described above, the BIG-IP appliance also allows Administrative-users access to objects to be further refined by the use of Administrative partitions. These logical storage units allows the Administrative-users to place certain BIG-IP objects within these repositories and then manage access by partitions in addition to user role. Once configured, this requires the user to have access to the partition in addition to their assigned role in order to access a configuration object within a given partition.

Access to Security Management functions is addressed through the PAM module functionality within the BIG-IP TMOS. Access is coordinated utilizing Role based access control mechanisms. Access to the Administrator Console is supported through a web GUI within the Apache web server (HTTPd) environment which is integrated with the underlying TMOS, or through tmsh over ssh.

A series of traffic management configuration options allow the authorized administrative-users to configure resources based on individual nodes, connection pools, and protocol based traffic profile settings which support the Information Flow Control requirements listed in Section FMT_MSA.1 Management of security attributes (2) and FMT_MSA.3 Static attribute initializationand enforced through the UNAUTHENTICATED and AUTHENTICATED security functional policies.

The TOE provides certificate management functions that allow Administrative-users to create X.509 self-signed certificates, import certificates signed by a Certificate Authority, and configure certificates for an SSL client in the Operational Environment. By installing the applicable certificates on the BIG IP, it allows the TOE to perform all of the SSL certificate verification functions on the traffic otherwise performed by either the client, the back-end server, or both.

Within the PAM module, rules are established for the creation of passwords assuring a minimum length, type, and lifetime for system passwords. Password policy is enforced through technical means for all users except for the Administrator and User Manager roles, where procedural compliance to this policy is required through administrator guidance.

1.8.4 Secure Communications

Secure Communication techniques are made available in the TOE for administrative-user access via SSL. An Administrative-user management port is provided for dedicated local access.

By default, the TOE uses uniquely generated 1024 bit RSA keys with self-signed certificates for securing communications with the web-based UI. An Administrative-user may generate new keys of 1024, 2048, or 4096 bits.

In addition, Administrative-users can access the TOE's command line utility, tmsh, securely via ssh. The TOE uses uniquely-generated 1024-bit RSA or DSA keys for securing this communication.

1.8.5 Secure Traffic

The TOE secures traffic using a hardware based security processor for SSL traffic, the software based TMM MicroKernel within the BIG-IP operating system for SSL handshaking, and an OpenSSL library for supporting local X.509 certificate verification.

To increase availability and capacity within the supported backend servers, the TOE may be configured to terminate SSL at the appliance. Through this function, called SSL Termination, the TOE establishes and terminates SSL traffic on behalf of the backend server pools using Client, Server, or both Client and Server Profiles with certificate based authentication requirements. When the optional feature, SSL termination with Client/Server verification, is selected within the TOE, certificate verification and revocation checks are executed externally leveraging an OCSP server in the Operational Environment.

SSL session persistence can be enabled based on configurable Client or Server SSL persistence profiles.

*Note: The cryptography used in this product has not been FIPS validated nor has it been analyzed or tested to conform to cryptographic standards during this evaluation. All cryptography has only been asserted as tested by the vendor.

1.8.6 Protection of TSF

Physical and logical protection of the TOE appliance is required to assure that TOE related security functions are not bypassed or altered. All access to the GUI based security management interface and tmsh requires identification and authentication.

The TOE provides logical separation between Administrator-user sessions and traffic domains to assure traffic flowing through the device to backend server resource cannot access TSF security management interfaces and configuration mechanisms.

TOE security functionality also protects against DoS attacks by limiting the numbers of TCP connect requests allowed within a given period of time.

The TOE allows Administrator-user access to TSF data through a dedicated management port through which a HTTPS session is established using a browser in the Operational Environment. These Administrative-user sessions are conducted over SSL and secured using AES encryption.

Alternatively, tmsh Administrative-user sessions are established over ssh and secured using AES encryption.

A comprehensive audit logging function assures that all Administrative-user authentication failures are logged providing a resource to determine if unauthorized personnel may be attempting to access TSF functions.

The Evaluated Configuration of the TOE is in the Redundant Pair configuration allowing for maximum availability under various failure conditions.

The TOE also preserves a secure state during various failure conditions and will transfer traffic and security enforcement activities to the redundant appliance in the event the primary appliance encounters a hardware or software failure.

1.8.7 User Data Protection: Information Flow Control

The BIG-IP mediates network traffic by evaluating traffic destined for backend server resources and routing it based on a series of configured checks and flow control rules. The TOE supports unauthenticated flows for any protocol and authenticated flows for HTTP and HTTPS. The TOE protects backend servers within the internal network from unauthorized or malicious traffic flows through packet deconstruction and analysis.

Information Flow Control policies are configured through security profiles in the TOE. The combination of configured security checks and routing rule enforcement assures that traffic is fully inspected and identified threats addressed (by discarding the traffic) prior to routing to resources in the Operational Environment.

BIG-IP traffic processing options include: SSL secure traffic, Content based compression of HTTP traffic, and Rules based Pool selection to assure highest availability and processing speed. Backend Servers are managed in resource Pools and flow control policies are deployed and enforced according to Pool memberships.

Packet filters enhance network security by specifying whether a BIG-IP system interface should accept or reject certain packets based on criteria that are configured by the applicable Administrative-user. Examples of criteria that can be configured into a packet filter rule are:

- The source IP address of a packet
- The destination IP address of a packet
- The destination port of a packet

As well as configuring individual packet filter rules, administrative users may configure global exemptions to specify “always allow” cases for certain objects, such as protocols, MAC addresses, IP addresses, and VLANs.

In addition, the TOE implements protocol analysis for HTTP, SMTP, and FTP traffic and through configuration of iRules scripts additional protocols may be analyzed. HTTPS traffic may be decrypted by the BIG-IP appliance and analyzed to HTTP security profiles.

The Protocol Security Module (component of the Traffic Management Module (TMM)) receives deconstructed HTTP packets from the TMM and performs a series of security checks to validate the traffic flow before allowing routing to backend server resources in the Operational Environment.

HTTP protocol and parameter length validation checks are conducted initially to assure that the request conforms to HTTP protocol standards such as RFC 2616 for HTTP version 1.1.

Following protocol validation, the TOE performs additional HTTP request parameter length checks to protect against buffer overflows, validates HTTP methods, and response codes. Headers are inspected to identify valueless headers and other factors that may be indicative of an HTTP Request Smuggling attack. A secondary level of security checking performed includes verifying coding methods for application attacks or command/code injection attempts that may also be designed to avoid detection. These coding methods (known as evasion techniques) could be indicative of a malicious traffic flow or access attempt. In addition, the TOE can be configured to enforce mandatory headers in support of backend application resources and validate files types against either a "files allowed" or "files disallowed" list that may be configured by the authorized Administrative-user.

SMTP security checks begin with a protocol compliance check to RFC 2821. The IP addresses or domain for the incoming mail is resolved with DNS A records to assure it's legitimate and the sender's identity is also checked against DNS SPF records (DNS TXT) to verify that the mail was allowed to be sending from the claimed domain. In addition, methods used with the SMTP traffic must not include disallowed methods configured within the security profile such as VRFY, EPN, or ETRN from the default security profile as well as those configured by the administrative user. The TOE may also be configured to reject the first received message and only validate the server upon a retry attempt, which usually indicates a legitimate source. The sender's identity and IP address and the recipient's identity are kept in memory until the timeout for the retry attempt expires.

Then, BIG-IP provides traffic shaping features that allow for the bandwidth throttling or rate limiting of traffic based on configured rules relating to IP, Protocol, or identified traffic attributes. In the event that SPAM senders are inadvertently allowed through these defenses, rate limiting of messages by sender is implemented to assure the configured maximum allowed number of messages cannot be exceeded by any single message source.

FTP traffic is analyzed by functionality provided by the Protocol Security Module. A protocol compliance check is performed to RFCs: 959, 1579, 3659 and security checks are performed on FTP traffic:

- Anonymous FTP requests
- Passive or active FTP requests
- FTP commands not included in the allowed list
- Command line length exceeds the maximum length allowed
- FTP logon attempts exceeds allowed number
- FTP traffic that fails protocol compliance checks

In the event a security rule for any of the protocol based listings above is triggered, the TOE can be configured to generate an alarm and/or route the traffic to a block response page or redirect to a configured location. Regardless, each time a security rule is matched (indicating violation) an audit log is generated in the Application (PSM) log.

The PSM module also generates violation data, statistics, and traffic reports for the traffic that the module inspects; allowing Administrative-users to evaluate security check results for malicious activity.

1.8.8 Email Alerts

Administrative-users with the administrator role with tmsh access may configure an email notification for eligible alerts. Email alerts may be set up on log events chosen by the administrative-user from a restricted list of events (with the exception of PSM events, which post an alert to the Statistics page of the GUI). When one of those selected log events is written to the log, an email is sent to the the configured email address.

1.9 Items Excluded from the TOE (security relevant)

This section identifies any security relevant items that are specifically excluded from the TOE:

1. Application templates – configurations restricted to manual approach (procedurally enforced).

2. The following modules, as they are separately licensed and not included in the CC Evaluated Configuration:

    a. BIG-IP Global Traffic Manager

    b. BIG-IP Link Controller

    c. BIG-IP Application Security Manager

    d. BIG-IP WebAccelerator System

    e. BIG-IP WAN Optimization Module

    f. BIG-IP Access Policy Manager

    g. BIG-IP Message Security Module

3. Application Security Policy Editor role, which is not included in a BIG-IP configuration except as part of the Application Security Module.

4. Always-On Management (AOM) – SSH access to AOM is disabled unless configured, and the Common Criteria evaluated configuration does not configure SSH for AOM. Serial console access to AOM is procedurally excluded from the Common Criteria Evaluated Configuration

5. bash shell – disabled by Appliance mode.

6. BigPipe Utility Command Line Interface (CLI) and Bigpipe Shell (bpsh) – deprecated in this release and therefore procedurally excluded. Note that:

    a. Users must not be created with the capability to access the bigpipe shell, either through the GUI or tmsh;

    b. The bigpipe shell must not be accessed though the tmsh “run /util bigpipe shell“ command;

    c. The bigpipe utility commands must not be accessed through the “run /util bigpipe

<command>" command.

7. SNMP (Remote Management of BIG-IP) is disabled via configuration script, therefore references to SNMP in the environment do not apply. However, email notification of alerts relies on modifying the alertd configuration file, which uses the snmptrap statement format to define the alert. References to snmptrap in that context do apply.

8. FIPS hardware, including hardware-based SSL offloading.

9. iSessions (relates to data center to data center deployment models) - requires BIG-IP® WAN Optimization Module which is not included in TOE.

10. Editing the configuration files specified in the TMOS Management Guide is excluded. The GUI or tmsh must be used for all system configuration.

11. IMI and VTY shells.

12. The abililty to configure the TOE via the appliance LCD display is disabled except during initial configuration.

13. Serial port.

14. Kerberos server. This is not enabled unless configured, and the Common Criteria evaluated configuration does not configure Kerberos. Note that the default Kerberos profile says that it is enabled, but without fully configuring the profile and attaching it to a virtual server, Kerberos itself is not configured and not usable. Thus, by default, Kerberos itself is not enabled.

15. iControl interface is procedurally excluded since all of the function it provides is also provided with the GUI and tmsh interfaces.

16. Note that since CRLs can quickly become outdated, their use and that of CRLDPs is excluded from the TOE. Therefore, an OCSP server is required in the Operational Environment for certificate revocation checks.

17. The following profiles (based on the list in the *Configuration Guide for BIG-IP Local Traffic Manager*, Chapter 5 (Understanding Profiles), section "Profile Types")

a. Services profiles: RTSP, Diameter, and iSession

b. Persistence profiles: Microsoft Remote Desktop

c. Protocol profiles: SCTP

d. SSL profiles: None are excluded

e. Authentication profiles: Kerberos Delegation

f. Other profiles: NTLM and Stream.

18. Protocol sanitization for protocols other than HTTP, FTP, and SMTP.

19. Ciphers other than those specified in Appendix A – Cryptographic Key Support in this document. Note that the CCMODE script described in the Guidance Wrapper (AGD) document changes the cryptographic defaults as they are described in guidance documents and supercedes those documents.

20. Cryptographic-related protocols other than SSHv2, SSLv3, and TLSv1.0.

21. Any features requiring root access to configure, since access to root is disabled via Appliance Mode. This includes, for example, Remote encrypted logging, since Appliance Mode precludes the ability to configure the SSH tunnel required for that function.
22. The gencert utility is excluded since it's only accessible though excluded shells. Key and certificate generation should be accomplished through the GUI instead.
23. References to CORBA, which is not used in the BIGIP.
24. TACACS+ is excluded as a remote authentication server.
25. Network boot
26. Software updates to the common-criteria evaluated configuration.
27. Batch mode tmsh transactions

1.10 Items not Evaluated (non-security relevant)

This section identifies aspects of the TOE that were not evaluated as part of the Common Criteria Evaluation. With the exception of those items listed as "separately licensed and not included with the TOE", items in this category include those features which may provide significant functional capability within the TOE and may be used by customers but are not security relevant.

Those items listed as "separately licensed and not included with the TOE" may have security-relevant aspects and should not be used with a Common Criteria evaluated configuration without careful review.

1. WebAccelerator™ Module (WAM) - separately licensed and not included with the TOE.
2. Link Controller (LC) - separately licensed and not included with the TOE.
3. Global Traffic Manager (GTM) - separately licensed and not included with the TOE.
4. Application Policy Module (APM) - separately licensed and not included with the TOE.
5. Enterprise Manager – separately licensed and not included with the TOE.
6. F5 Management Pack – separately licensed and not included with the TOE.
7. Advanced Routing – separately licensed and not included with the TOE.
8. Optimization of network and application traffic; load balancing.
9. HTTP compression.
10. Caching.
11. Aggregation of client requests.
12. Routing around slower or degraded routes.
13. Selective data compression.
14. Windows NT LAN Manager authentication protocol (NTLM). The BIG-IP passes this protocol through, but does not itself perform NTLM authentication.
15. Network resource monitoring.

16. Trunk (link aggregation).
17. Spanning Tree Protocols
18. Network Tunnels
19. Bigtop utility – this utility provides statistical monitoring only.
20. SNAT – "Source NAT". BIG-IP implements SNAT as mapping a source client IP address to a translation address defined on the BIG-IP system.
21. Booting from different volumes. The BIG-IP may be configured with multiple volumes but only booting from the slot containing the Common Criteria-evaluated configuration is recommended.

2 Conformance Claims

The TOE is conformant with Common Criteria (CC) Version 3.1, revision 3, Part 2 Extended.

The TOE is Common Criteria (CC) Version 3.1, revision 3, Part 3 conformant at EAL 2 + augmented ALC_FLR.2.

The TOE is compliant with International Interpretations with effective dates on or before June 18, 2010.

The TOE does not claim compliance with a specific Protection Profile.

3 Security Problem Definition

This section contains assumptions regarding the security environment and the intended usage of the TOE and threats on the TOE and the Operational Environment.

3.1 Assumptions

This section contains assumptions regarding the security environment and the intended usage of the TOE.

A.PHYSEC The TOE is physically secure.

A.GENPUR There are no general-purpose computing capabilities (e.g., the ability to execute arbitrary code or applications) and storage repository capabilities on the TOE.

A.PUBLIC The TOE does not host public data.

A.NOEVIL Administrative-users are non-hostile and follow all administrator guidance; however, they are capable of error.

A.SINGEN The TOE is configured and connected such that information cannot flow among the internal and external networks unless it passes through the TOE.

A.DIRECT Human users within the physically secure boundary protecting the TOE may attempt to access the TOE from some direct connection (e.g., a console port).

A.OCSP An OSCP Server will be available in the Operational Environment for the purpose of verifying the revocation status of certificates on behalf of the TOE.

A.SYSLOG If remote syslog is used in the operational environment, the remote syslog server and its connection to the TOE are physically secure.

3.2 Threats Addressed by the TOE

The following identifies threats to the TOE that may be indicative of vulnerabilities in or misuse of IT resources. The threat agents are either unauthorized persons or external IT entities not authorized to use the TOE itself.

T.NOAUTH An unauthorized person may attempt to bypass the security of the TOE to configure the TOE maliciously or in an insecure mannor, allowing access to security functions and/or non-security functions provided by the TOE.

T.REPLAY An unauthorized person may use valid identification and authentication data obtained to access functions provided by the TOE, allowing them to maliciously or insecurely configure the TOE or interfere with the authorized-user-designated operation of the TOE.

T.ASPOOF An unauthorized person on an external network may attempt to bypass the information flow control policy by disguising authentication data (e.g., spoofing the source address) and masquerading as a legitimate user or entity on an internal network. This gives the unauthorized person illicit access to backend servers, and may result in information disclosure to unauthorized users or

tampering by those users.

T.DATAFLOW	An unauthorized person may tamper with security properties associated with data passing through the TOE, corrupting or removing them. This may lead to illict access to backend servers, resulting in data disclosure to unauthorized parties or tampering by those parties.
T. DISCLOSE	Details about the internal network and backend server environment may be inadvertently disclosed which could be used by a malicious user to formulate an attack on the TOE or internal network resources.
T.MEDIAT	An unauthorized person may send impermissible information through the TOE, which results in the exploitation of resources on the internal network.
T.OLDINF	An unauthorized person may gather residual information from a previous information flow or internal TOE data by monitoring the padding of the information flows from the TOE, Gathering residual information from previous information flows results in information disclosure of backend server data to unauthorized parties; gathering residual information from internal TOE data may result in unauthorized persons having access to data that allows them to gain access to or otherwise affect TOE configuration and operation.
T.PROCOM	An unauthorized person or unauthorized external IT entity may be able to view, modify, and/or delete security related information that is sent between an authorized administrative-user and the TOE, resulting in potentially-insecure TOE configurations that can later be exploited, or configurations that render the TOE non-operational or operational in an unintended fashion.
T.AUDACC	Persons may not be accountable for the actions that they conduct because the audit records are not reviewed, thus allowing both the integrity of the TOE to be threatened and an attacker to escape detection.
T.SELPRO	An unauthorized person may read, modify, or destroy security critical TOE configuration data, resulting in potentially-insecure TOE configurations that can later be exploited, or configurations that render the TOE non-operational or operational in an unintended fashion.
T.TOE_FAIL	Hardware component failure or software bugs can cause the failure of a TOE appliance and may result in loss of traffic and/or failure to meet the TSF.
T.UNATTENDED	A malicious user may access the TSF through an unattended Administrative-user session via the GUI with the TOE, resulting in malicious or insecure TOE configurations that can later be exploited, or configurations that render the TOE non-operational or operational in an unintended fashion..
T.RESOURCE_X	A malicious process or user may block others from TOE system resources (e.g., connection state tables) via a resource exhaustion denial of service attack.
T.RULE_LIMIT	An unauthorized traffic flow or malicious content may traverse the TOE and compromise backend server resources. This is caused by the inability

for Administrative-Users to establish customized flow control policies suitable for their particular deployment scenario and threat profile. That is, Administrative-users cannot define a needed customized rule and thus malicious or unauthorized traffic that would otherwise be blocked by that rule is not blocked and backend server resources are compromised.

3.3 Threat to be Addressed by Operating Environment

The threat possibility discussed below must be countered by procedural measures and/or administrative methods.

T.TUSAGE The TOE may be inadvertently configured, used, and administered in an insecure manner by either authorized or unauthorized persons.

T.AUDMOD A malicious user may modify the audit records sent from the TOE to a remote syslog server, causing persons not to be held accountable for their actions.

T.UNATTENDED A malicious user may access the TSF through an unattended Administrative-user session via the CLI with the TOE, resulting in malicious or insecure TOE configurations that can later be exploited, or configurations that render the TOE non-operational or operational in an unintended fashion..

3.4 Organizational Security Policies

There are no Organizational Security Policies applicable to this security target.

4 SECURITY OBJECTIVES

This section describes the security objectives for the TOE and the TOE's operating Environment. The security objectives are divided between TOE Security Objectives and Security Objectives for the Operating Environment.

4.1 Security Objectives for the TOE

The following are the IT security objectives for the TOE and are satisfied by technical means by hardware/software:

O.IDAUTH	The TOE must uniquely identify and authenticate the claimed identity of all users, before granting a user access to the TOE security management interface or, for certain specified services (i.e.: SSL), to a connected network. The TOE supports the use of X.509 certificates and password based authentication.
O.MEDIAT	The TOE must mediate the flow of all information between clients and servers located on internal and external networks governed by the TOE, and must ensure that memory locations used to pad packets does not release information from previous sessions.
O.SECSTA	Upon initial start-up of the TOE or recovery from an interruption in TOE service, the TOE must not compromise its resources or those of any connected network.
O.CRYPTO	The TOE must perform encryption/decryption to support protection of authentication data, security related information and data in transit with the exception of audit records sent to a remote syslog server.
O.SELPRO	The TOE must protect itself against attempts by unauthorized users to bypass, deactivate, or tamper with TOE security functions.
O.AUDREC	The TOE must provide a means to assure that users are accountable for their actions.
O.AUDALERT	The TOE must provide a means to notify the TOE Administrative-user and implement Administrative-user specified actions for all identified conditions.
O.ACCOUN	The TOE must provide user accountability for information flows through the TOE and for authorized administrative-user use of security functions related to audit.
O.SECFUN	The TOE must provide all the functions and facilities necessary to support the administrative-users in their management of the security of the TOE, and restrict these functions and facilities from unauthorized use.
O.SAFE_FAIL	The TOE will protect the TSF in the event of all failure conditions and preserve correct operations during specified failure events.
O.LIMEXT	The TOE must provide the means for an authorized administrative-user to control and limit access to TOE security functions by an authorized external IT entity.

O.PROXY	The TOE must proxy connections made through it in such a way as to mask backend server resources and associated IP addresses while preserving the security properties of data flows through the appliance.
O.RESOURCE_X	The TOE must provide mechanism to identify and thwart DoS attempts on the appliance.
O.SESSION_TERM	The TOE must provide mechanisms that terminate unattended administrative-user GUI session after a configured period of inactivity and must allow administrative-users to terminate open sessions.
O.SCRIPT_FUNC	The TOE must provide a programmable script based mechanism to develop custom flow control polices, perform traffic analysis, and implement specified actions based on programmatic constructs and conditional statements.

4.2 Security Objectives for the Environment

The following security objectives apply to the Operational Environment. The following are imposed through technical resources in the Operational Environment:

OE.OCSP	The Operational Environment will provide an OSCP Server in the Operational Environment for the purpose of verifying the revocation status of certificates on behalf of the TOE.

These non-IT security objectives, in addition to corresponding assumptions, are to be satisfied without imposing technical requirements on the TOE. These objectives are satisfied through the application of procedural or administrative measures:

OE.PHYSEC	The TOE is physically secure.
OE.GENPUR	There are no general-purpose computing capabilities (e.g., the ability to execute arbitrary code or applications) and storage repository capabilities on the TOE.
OE.PUBLIC	The TOE does not host public data.
OE.NOEVIL	Authorized administrative-users are non-hostile and follow all administrator guidance; however, they are capable of error.
OE.SINGEN	Information cannot flow among the internal and external networks unless it passes through a TOE appliance.
OE.DIRECT	Human users within the physically secure boundary protecting the TOE may attempt to access the TOE from some direct connection (e.g., a console port) if the connection is part of the TOE.
OE.REMACC	Authorized administrative-users may access the TOE remotely from the internal and external networks.
OE.GUIDAN	The TOE must be delivered, installed, administered, and operated in a manner that maintains security.

OE.ADMTRA	Authorized administrative-users are trained as to establishment and maintenance of security policies and practices.
OE.SYSLOG	If remote syslog is used in the operational environment, the remote syslog server and its connection to the TOE must be made physically secure.

4.3 Mapping of Threats to Security Objectives

The following table represents a mapping of the threats to the IT Security Objectives defined in this ST.

	T.NOAUTH	T.REPLAY	T.ASPOOF	T.MEDIAT	T.OLDINF	T.PROCOM	T.AUDACC	T.DATAFLOW	T.DISCLOSE	T.SELPRO	T.RESOURCE_X	T.TOE_FAIL	T.UNATTENDED	T.RULE_LIMIT	T.AUDMOD
O.IDAUTH	X														
O.MEDIAT			X	X	X										
O.SECSTA	X									X					
O.CRYPTO	X					X									
O.SELPRO	X									X					
O.AUDALERT				X							X				
O.AUDREC							X								
O.ACCOUN							X								
O.SECFUN	X	X													
O.LIMEXT	X														
O.PROXY								X	X						
O.RESOURCE_X											X				
O.SAFE_FAIL												X			
O.SESSION_TERM													X		
O.SCRIPT_FUNC														X	
O.SYSLOG															X

Table 4: Summary of Mappings between Threats and IT Security

4.4 Rationale for IT SECURITY OBJECTIVES

O.IDAUTH	This security objective is necessary to counter the threat: T.NOAUTH because it requires that users be uniquely identified before accessing the TOE.
O.MEDIAT	This security objective is necessary to counter the threats: T.ASPOOF, T.MEDIAT, and T.OLDINF which have to do with getting impermissible information to flow through the TOE. This security objective requires that all information that passes through the networks is mediated by the TOE and that no residual information is transmitted.

O.SECSTA	This security objective ensures that no information is compromised by the TOE upon start-up or recovery and thus counters the threats: T.NOAUTH and T.SELPRO.
O.CRYPTO	This security objective is necessary to counter the threats and policy: T.NOAUTH and T.PROCOM by requiring that an authorized administrative-user use encryption when performing administrative functions on the TOE remotely.
O.SELPRO	This security objective is necessary to counter the threats: T.SELPRO, and T.NOAUTH because it requires that the TOE protect itself from attempts to bypass, deactivate, or tamper with TOE security functions.
O.AUDREC	This security objective is necessary to counter the threat: T.AUDACC by requiring a readable audit trail and a means to search and sort the information contained in the audit trail.
O.AUDALERT	This security objective is necessary to counter the threats: T. MEDIATE and T.RESOURCE_X as it provides the ability to detect a security violation based on configured criteria and takes specified action to mitigate the event and/or alert administrative-users.
O.ACCOUN	This security objective is necessary to counter the threat: T.AUDACC because it requires that users are accountable for information flows through the TOE and that authorized administrative-users are accountable for the use of security functions related to audit.
O.SECFUN	This security objective is necessary to counter the threats: T.NOAUTH, T.REPLAY by requiring that the TOE provide functionality that ensures that only the authorized administrative-user has access to the TOE security functions.
O.LIMEXT	This security objective is necessary to counter the threat: T.NOAUTH because it requires that the TOE provide the means for an authorized administrative-user to control and limit access to TOE security functions.
O.PROXY	This security objective in necessary to counter the threat: T.DATAFLOW and T.DISCLOSE because it requires that the TOE masks backend server resources and associated IP addresses while preserving the security properties of data flows through the appliance.
O.RESOURCE_X	This security objective is required to counter the threat: T.RESOURCE_X as it provides mechanisms within the TOE to identify and thwart DoS attacks.
O.SAFE_FAIL	This security objective is required to counter the threat: T. TOE_FAIL as it provides mechanisms that preserve correct TSF operations and the TSF in the event of a failure of a single appliance in a redundant pair configuration.

O.SESSION_TERM	This security objective is required to counter the threat: T.UNATTENDED as it provides mechanisms that terminate unattended sessions after a configured period of inactivity and allows administrative-users to terminate their session with the TOE.
O.SCRIPT_FUNC	This security objective is required to counter the threat: T.RULE_LIMIT as it provides a scripting mechanism that allows the development of scripts by administrative-users for the purpose of implementing customized flow control policies and invoking TSF functions.

4.5 RATIONALE FOR SECURITY OBJECTIVES FOR THE ENVIRONMENT

The following security objectives for the environment are, in part, a re-statement of the security assumptions, those security objectives trace to all aspects of the assumptions. Therefore they are not mapped in a table.

OE.PHYSEC	The TOE is physically secure.
OE.GENPUR	There are no general-purpose computing capabilities (e.g., the ability to execute arbitrary code or applications) and storage repository capabilities on the TOE.
OE.PUBLIC	The TOE does not host public data.
OE.NOEVIL	Authorized administrative-users are non-hostile and follow all administrator guidance; however, they are capable of error.
OE.SINGEN	Information cannot flow among the internal and external networks unless it passes through a TOE appliance.
OE.DIRECT	Human users within the physically secure boundary protecting the TOE may attempt to access the TOE from some direct connection (e.g., a console port) if the connection is part of the TOE.
OE.REMACC	Authorized administrative-users may access the TOE remotely from the internal and external networks.
OE.OCSP	This security objective provides that an OSCP Server in the Operational Environment for the purpose of verifying the revocation status of certificates on behalf of the TOE.
OE.SYSLOG	This objective is necessary to counter the threat T.AUDMOD. If logging to a remote syslog server is configured in the operational environment, this objective ensures that the log records cannot be modified in transit between the TOE and the remote server. This is necessary because it is not possible to configure the TOE to encrypt that connection in Appliance Mode.

The following security objectives are not restatements of their associated threats and therefore a rationale is provided below:

OE.GUIDAN — This non-IT security objective is necessary to counter the threat: T.TUSAGE and T.AUDACC because it requires that those responsible for the TOE ensure that it is delivered, installed, administered, and operated in a secure manner. This objective also counters the threat T.UNATTENDED since guidance is given to administrative-users to close their CLI sessions before leaving their workstation (an aspect of administered in a secure manner).

OE.ADMTRA — This non-IT security objective is necessary to counter the threat: T.TUSAGE and T.AUDACC because it ensures that authorized administrative-users receive the proper training.

	T.TUSAGE	**T.AUDACC**	**T.UNATTENDED**
OE.GUIDAN	X	X	X
OE.ADMTRA	X	X	

Table 5: Summary of Mappings between Threats and Security Objectives for the Environment

5 Extended Components Definition

5.1.1 FPT_FLS_EXP – Failure with Preservation of Secure Configuration State

Family Behavior

The requirements of this family ensure that the TOE will always enforce its SFRs in the event of identified categories of failures in the TSF.

Component Leveling

This family consists of only one component, FPT_FLS_EXP.1 Failure with preservation of secure configuration state, which requires that the TSF provide an interface to preserve a secure configuration state.

Management: FPT_FLS_EXP.1

There are no management activities foreseen.

Audit: FPT_FLS_EXP.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) Basic: Failure of the TSF.

5.1.1.1 FPT_FLS_EXP.1 Failure with Preservation of Secure Configuration State

Hierarchical to: None

Dependencies: None

FPT_FLS_EXP.1.1 **The TSF shall provide [assignment: *authorized users*] an interface to preserve a secure configuration state.**

5.1.2 FMT_SCR_EXP – Scripting of Flow Control Rules

Family Behavior

This Family defines the requirement for the TSF to provide to Administrative-users the ability to create scripts through the security management interface that can implement conditional flow control functions, including flow control rules for the purpose of implementing the applicable flow control SFP in unique ways based on the needs of the individual deployment and security objectives.

Component Leveling

At FMT_SCR_EXP.1 Scripting of Flow Control rules, the TSF shall provide a scripting utility for the purpose of creating custom flow control rules and flow control functions.

Management: FMT_SCR_EXP.1

The following actions could be considered for the management functions in FMT:

a. Managing the roles that can create flow control rules/functions using the scripting utility.

Audit: FMT_SCR_EXP.1

The following actions should be auditable if FAU_GEN.1 Security audit data generation is included in the PP/ST:

a. Creation of a Script file

b. Deletion of a Script file

5.1.2.1 FMT_SCR_EXP.1 Scripting of Flow Control Rules

Hierarchical to: None

Dependencies: FMT_MSA.1, FDP_IFF.1

FMT_SCR_EXP.1.1	The TSF shall support the development of custom flow control rules using a scripting language for use in constructing a rules set.
FMT_SCR_EXP.1.2	The TSF shall provide a Scripting language which includes the following constructs at a minimum [selection: Event Declarations, Operators, Commands] [assignment: other programmatic constructs] with syntax based on the [assignment: (applicable standard defining syntax)] standard.
FMT_SCR_EXP.1.3	The TSF shall support the flow control rules scripted in FMT_SCR.1.1 to be used in implementing the following Flow Control Security Functional Policy(s) [assignment: SFP].
FMT_SCR_EXP.1.4	The TSF shall implement the scripted action in the event the conditional rule set statements are TRUE and shall not implement the scripted action in the event the conditional rule set statements are FALSE in accordance with the [assignment: Flow Control SFP] defined in FDP_IFF.1.
FMT_SCR_EXP.1.5	The TSF shall support implementation of the following functions using the scripting language command set [assignment: list of traffic management related functions supported by the scripting language command set].

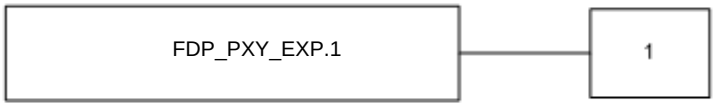
5.1.3 FDP_PXY_EXP – Reverse Proxy

Family Behavior

This Family defines the requirement for the TSF to provide Reverse Proxy functionality to

prevent internal network entities from disclosure of network topology, routing addressing and server resource identification to external networks.

Component Leveling



At FDP_PXY_EXP.1 Reverse Proxy, the TSF shall rewrite packet headers and other identifying information on packets traversing through the TOE from the internal to the external networks.

Management: FDP_PXY_EXP.1

The following actions could be considered for the management functions in FMT:

a) Configuration of internal IPs to external translation address.

b) Configuration of external translation addresses to internal IPs.

Audit: FDP_PXY_EXP.1

There are no auditable events foreseen.

5.1.3.1 FDP_PXY_EXP.1 Reverse Proxy

Hierarchical to: None

Dependencies: None

FDP_PXY_EXP.1.1	The TSF shall rewrite [assignment: ***packet data***] for [assignment: ***transport/internet layer protocol***] packets traversing the TOE between internal and external networks.
FDP_PXY_EXP.1.2	The TSF shall rewrite Inbound [assignment: ***packet data***] from external addresses to internal addresses as configured by the [assignment: ***the authorised identified roles***].
FDP_PXY_EXP.1.3	The TSF shall rewrite [assignment: ***packet data***] from internal addresses to external IP Addresses as configured by the [assignment: ***the authorised identified roles***].
FDP_PXY_EXP.1.4	The TSF shall checksum packet headers by generating and verifying a [assignment: ***checksum size in bits***] CRC per [assignment: ***list of standards***].
FDP_PXY_EXP.1.5	The TSF shall create discrete buffers for each packet, associate the buffers to a data flow and process each flow separately in a single threaded fashion

5.1.4 FAU_SAA_EXP – Potential Violation Analysis

Family Behavior

This family defines requirements for automated means that analyse system activity and audit data looking for possible or real security violations. This analysis may work in support of intrusion detection, or automatic response to a potential security violation.

Component Leveling

In FAU_SAA_EXP.1 Potential violation analysis, basic threshold detection on the basis of a fixed rule set is required.

Management: FAU_SAA_EXP.1

The following actions could be considered for the management functions in FMT:

a) maintenance of the rules by (adding, modifying, deletion) of rules from the set of rules.

Audit: FAU_SAA_EXP.1

There are no auditable events required.

5.1.4.1 FAU_SAA_EXP.1 Potential Violation Analysis

Hierarchical to: No other components

Dependencies: None

FAU_SAA_EXP.1.1 The TSF shall be able to apply a set of rules in monitoring user traffic and based upon these rules indicate a potential violation of the enforcement of SFRs.

FAU_SAA_EXP.1.2 The TSF shall enforce the following rules for monitoring user traffic:

a. Accumulation or combination of [assignment: ***subset of defined events****]* known to indicate a potential security violation*;*

b. [assignment: *any other rules*].

5.1.5 FIA_SOS_EXP – Specification of Secrets

Family Behavior

This family defines requirements for mechanisms that enforce defined quality metrics on provided secrets and generate secrets to satisfy the defined metric.

Component Leveling

FIA_SOS.1 Verification of secrets, requires the TSF to verify that secrets meet defined quality metrics.

Management: FIA_SOS_EXP.1

The following actions could be considered for the management functions in FMT:

a. the management of the metric used to verify the secrets.

Audit: FIA_SOS_EXP.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a. Minimal: Rejection by the TSF of any tested secret;

b. Basic: Rejection or acceptance by the TSF of any tested secret;

c. Detailed: Identification of any changes to the defined quality metrics.

5.1.5.1 FIA_SOS_EXP.1 Verification of Secrets

Hierarchical to: No other components

Dependencies: None

FIA_SOS_EXP.1.1 The TSF shall provide a mechanism to verify that [assignment: ***list of secrets***] meet [assignment: ***list of rules enforced on secrets***].

5.1.6 FTA_SSL_EXP – Session Locking and Termination

Family Behavior

This family defines requirements for the TSF to provide the capability for TSF-initiated and user-initiated locking, unlocking, and termination of interactive sessions.

Component Leveling

FTA_SSL.3 TSF-initiated termination, provides requirements for the TSF to terminate the session after a specified period of user inactivity.

Management: FTA_SSL_EXP.3

The following actions could be considered for the management functions in FMT:

a. specification of the time of user inactivity after which termination of the interactive session occurs for an individual user;

b. specification of the default time of user inactivity after which termination of the interactive session occurs.

Audit: FTA_SSL_EXP.3

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a. Minimal: Termination of an interactive session by the session locking mechanism.

5.1.6.1 FTA_SSL_EXP.3 TSF_Initiated Termination

Hierarchical to: No other components

Dependencies: None

FTA_SSL_EXP.3.1 The TSF shall terminate an interactive [assignment: *TSF interface*] session after a [assignment*: time interval of user inactivity*].

5.1.7 FRU_RSA_EXP – Maximum Quotas

Family Behavior

The requirements of this family allow the TSF to control the use of resources by users and subjects such that denial of service will not occur because of unatuhorised monopolization of resources.

Component Leveling

Management: FRU_RSA_EXP.1

The following actions could be considered for the management functions in FRU:

a. Specifying maximum limits for a resource for groups and/or individual users and/or subjects by an administrator.

Audit: FRU_RSA_EXP.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a. Minimal: Rejection of allocation operation due to resource limits.

b. Basic: All attempted uses of the resource allocation functions for resources that are under control of the TSF.

5.1.7.1 FRU_RSA_EXP.1 Maximum Quotas

Hierarchical to: No other components

Dependencies: None

FRU_RSA_EXP.1.1 The TSF shall enforce maximum quotas of the following resources: [assignment; *controlled resources*] that [selection: *individual user, defined groups of users, subjects, the TOE*] can use [selection: *simultaneously, over a specified period of time*].

6 Security Requirements

The security requirements that are levied on the TOE are specified in this section of the ST. These security requirements are defined in Sections 6.2

6.1 Conventions

The CC defines four operations on security functional requirements. The conventions below define the conventions used in this ST to identify these operations. Note: combinations of these conventions represent text where multiple operations were performed

[Assignment]: indicated with [ ] – brackets

Selection: indicated with *italicized* text

Refinement: ***additions indicated with bold text and italics***

deletions indicated with strike-through ~~bold text and italics~~

Iteration: indicated with typical CC requirement naming followed by a number following the description for each iteration (e.g., FMT_MSA.1 Management of security attributes (2))

TOE Security Functional Requirements	
FAU_ARP.1	Security Alarms
FAU_GEN.1	Audit data generation
FAU_GEN.2	User identity association
FAU_SAA.1	Potential Violation Analysis
FAU_SAA_EXP.1	Potential Violation Analysis
FAU_SAR.1	Audit review
FAU_SAR.3	Selectable audit review
FAU_STG.1	Protected audit trail storage
FAU_STG.3	Action in case of possible audit data loss
FCS_CKM.1(1)	Cryptographic Key Generation-Asymmetric Keys
FCS_CKM.1(2)	Cryptographic Key Generation-Symmetric Keys
FCS_CKM.1(3)	Cryptographic Key Generation – PBKDF-generated Keys
FCS_COP.1(1)	Cryptographic Operation-Secure Traffic
FCS_COP.1(2)	Cryptographic Operation-Administrative-user Sessions
FCS_COP.1(3)	Cryptographic Operation – Cookie Encryption
FCS_COP.1(4)	Cryptographic operation - RSA signature/verification
FCS_COP.1(5)	Cryptographic operation - Hashing
FCS_COP.1(6)	Cryptographic operation - SSH

FCS_COP.1(7)	Cryptographic operation – Internal TSF data transfer
FCS_COP.1 (8)	Cryptographic operation – DSA signature/verification
FIA_ATD.1	User attribute definition – Administrative-users
FIA_UID.2	User identification before any action
FIA_UAU.2	User authentication before any action
FIA_UAU.5	Multiple authentication mechanisms
FIA_AFL.1	Authentication failure handling
FIA_SOS_EXP.1	Verification of Secrets
FMT_MSA.1(1)	Management of security attributes (1)
FMT_MSA.1(2)	Management of security attributes (2)
FMT_MSA.3 (1)	Static attribute initialization (1)
FMT_MSA.3 (2)	Static attribute initialization (2)
FMT_MSA.3 (3)	Static attribute initialization (3)
FMT_MTD.1(1)	Management of TSF data (1)
FMT_MTD.1(2)	Management of TSF data (2)
FMT_MTD.1(3)	Management of TSF data (3)
FMT_MTD.1(4)	Management of TSF data (4)
FMT_MTD.1(5)	Management of TSF data (5)
FMT_MTD.1(6)	Management of TSF data (6)
FMT_MTD.1(7)	Management of TSF data (7)
FMT_SAE.1	Time-limited authorization
FMT_SMF.1	Specification of Management Functions
FMT_SMR.1	Security roles
FMT_MOF.1	Management of Security Functions behavior
FPT_FLS.1	Failure with preservation of secure state
FPT_FLS_EXP.1	Failure with preservation of secure configuration state
FPT_ITC.1	Inter-TSF confidentiality during transmission
FPT_ITI.1	Inter-TSF detection of modification
FPT_STM.1	Reliable time stamps
FDP_IFC.1(1)	Subset information flow control-unauthenticated
FDP_IFC.1(2)	Subset information flow control-authenticated
FDP_IFF.1(1)	Simple security attributes-unauthenticated
FDP_IFF.1(2)	Simple security attributes-authenticated
FDP_RIP.1	Subset residual information protection
FTA_SSL_EXP.3	TSF Initiated Termination
FTA_SSL.4	User-Initiated Termination
FRU_FLT.1	Degraded fault tolerance

FRU_PRS.1	Limited priority of service
FTP_ITC.1	Inter-TSF trusted channel
FRU_RSA_EXP.1(1)	Maximum quotas (1) -TCP Connections
FRU_RSA_EXP.1(2)	Maximum quotas (2)-TCP SYN queue entries
FMT_SCR_EXP.1	Scripting of Flow Control Rules
FDP_PXY_EXP.1	Reverse Proxy

Table 6: Functional Requirements

6.2 TOE Security Functional Requirements

6.2.1 SECURITY AUDIT (FAU)

6.2.1.1 FAU_ARP.1 Security alarms

FAU_ARP.1.1 The TSF shall take [the following action: alert the configured Administrative-user via email (if configured) except for PSM-detected potential security violations; if a PSM-detected potential security violation, post to the Statistics screen of the GUI; if email is not configured, no action] upon detection of a potential security violation.

6.2.1.2 FAU_GEN.1 Audit data generation

FAU_GEN.1.1 The TSF shall be able to generate an audit record of the following auditable events:

a) Start-up and shutdown of the ***"command function audit log" type*** audit functions;

b) All auditable events for the *not specified* level of audit**;** and

c) [the events listed in Table 7].

FAU_GEN.1.2 The TSF shall record within each audit record at least the following information:

a) Date and time of the event, type of event, subject identity, outcome (success or failure) of the event; and

b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, [information specified in column three of Table 7].

Description	**Auditable Event**	**Additional Audit Record Content**
Local Traffic Events Log type: Local Traffic Event logs	• IP packet discard events due to exceptional circumstances or invalid parameters (such as a bad checksum) • MCP/TMM configuration events • Pool and node status change events	Timestamp, Host Name, Description, Service and Status Code information

	• Network events (layer 1) • iRules script events related to run-time iRules script processing, when specified in the iRule • General TMM events such as TMM startup, shutdown, and failover • Tcpdump startup and shutdown	
FMT_SMR.1 Log Type: Audit Log (Command audit functions)	Modifications to the configured roles.	Username, Transaction ID, Event description
FIA_UID.2 Log Type: Audit Log (System)	Use of the user identification mechanism.	Username, Transaction ID, Event description
FIA_UAU.2 Log Type: Audit Log (System)	Use of the authentication mechanism.	Username, Transaction ID, Event description
FIA_UAU.5 Log Type: Audit Log (System)	Use of the authentication mechanism	Username, Transaction ID, Event description
FIA_AFL.1 Log Type: Audit Log (System)	Passing the threshold for unsuccessful authentication attempts	Username, Transaction ID, Event description
FMT_MOF.1 Log Type: Audit Log (Command audit functions)	Use of the following functions listed in this requirement. ▪ PSM Security Profile Management ▪ Backup and Restore for TSF configuration data ▪ Enabling/Disabling of command audit functions ▪ Startup/Shutdown TOE operations[2] ▪ Proxy Translation Addresses configuration (SNAT) ▪ Query Audit Logs ▪ Default SFP security Attributes ▪ Virtual Server Management ▪ Administrative-user account policy management (including Password Policy, Authentication Failure configuration, and Session Timeout)	Username, Transaction ID, Event description

[2] TOE startup and shutdown operations are logged via: auditing the command used to bring down the system (e.g. tmsh reboot), placing a boot marker audit record when the system boots up, and auditing the configuration file reload. The combination of these events allows the administrative user to identify gaps in audit logs caused by a system shutdown / startup.

	▪ Administrative-user account management (except changing own password) ▪ Change own password ▪ View own user account information ▪ Node/Pool configuration ▪ Protocol profile configuration ▪ iRules script configuration ▪ Monitor ▪ Enable/Disable Nodes and Pool Members ▪ Authentication Profile configuration ▪ SSL Profile Configuration ▪ Key and Certificate management ▪ Syslog server configuration ▪ OCSP server configuration ▪ Memory protection configuration ▪ Partition Management	
FMT_SAE.1 Log Type: Audit Log (Command audit functions)	Setting of maximum duration for passwords	Username, Transaction ID, Event description
FMT_SCR_EXP.1 Log Type: Audit Log (Command audit functions)	Creation, Deletion of iRules scripts	Username, Transaction ID, Event description
Application Security Events Log Type: Application Security (PSM) Logs	Events relating to implementation and triggering of Application Security Profile events related to the Protocol Security Module	Timestamp, Host Name, Violation type, Service and Status; and Support ID for http violations
Packet Filter Events Log type: Packet Filter Event Logs	Packet filter messages that result from the operation of packet filter log rules.	Timestamp, Host Name, Description, Service and Status Code information
System Events Log type: System Event Logs	System event messages • Audit log warning	Timestamp, Host Name, Description, and Service

Table 7: Events logged by BIG-IP

6.2.1.3 FAU_GEN.2 User identity association

FAU_GEN.2.1 For audit events resulting from actions of identified users, the TSF shall be able to associate each auditable event with the identity of the user that caused the event.

6.2.1.4 FAU_SAR.1 Audit review

FAU_SAR.1.1 The TSF shall provide [Administrator] with the capability to read [all audit

trail data] from the audit records.

FAU_SAR.1.2 The TSF shall provide the audit records in a manner suitable for the user to interpret the information.

6.2.1.5 FAU_SAR.3 Selectable audit review

FAU_SAR.3.1 The TSF shall provide the ability to perform *searches and sorting* of audit data based on *critera for each log as specified in* Table 8: Log searching and sorting..

		GUI				
Parameter	**Tmsh Search[3]**	**System**	**Packet Filter**	**Local Traffic**	**Audit**	**Application Security**
User identity (hotname / username)	grep[4]	n/a	Search	Search	Search	n/a
Dates	Search	Sort	Sort	Sort	Search / Sort	Sort
Times	grep	Sort	Sort	Sort	Search / Sort	Sort
Addresses	grep	n/a	Search	n/a	Search	Search
Keyword filter	grep	Search	Search	Search	Search	Search
Log level	grep	Sort	Search / Sort	Sort	n/a	Sort
Service Transaction #	grep	n/a	n/a	n/a	Sort	n/a
Service	grep	Sort	n/a	Sort	n/a	Sort
Session ID	grep	n/a	Sort	n/a	n/a	n/a
Status code	grep	n/a	Sort	Sort	n/a	Sort
Event	grep	Search / Sort	Search / Sort	Search / Sort	Search / Sort	Search / Sort

Table 8: Log searching and sorting

6.2.1.6 FAU_SAA.1 Potential Violation Analysis

FAU_SAA.1.1 The TSF shall be able to apply a set of rules in monitoring the audited events and based upon these rules indicate a potential violation of the enforcement of SFRs.

FAU_SAA.1.2 The TSF shall enforce the following rules for monitoring audited events:

[3] Tmsh does not have a sort option for logs.

[4] The tmsh show log command allows display by number of lines and by date ranges; all other searches must be done by piping the log through grep.

a) Accumulation or combination of [audited events described in Table 7: Events logged by BIG-IP] known to indicate a potential security violation;

b) [No other rules].

6.2.1.7 FAU_SAA_EXP.1 Potential Violation Analysis (Explicit)

FAU_SAA_EXP.1.1 The TSF shall be able to apply a set of rules in monitoring user traffic and based upon these rules indicate a potential violation of the enforcement of SFRs.

FAU_SAA_EXP.1.2 The TSF shall enforce the following rules for monitoring user traffic:

a) Accumulation or combination of [SYN connects (SYN flood DoS attack Threshold Activation) (max = 16384)] known to indicate a potential security violation;

b) [HTTP security profile violations:
- HTTP protocol compliance violation (i.e.: RFC2616, HTTP/1.1)
- matching of disallowed HTTP coding methods
- triggering a Evasion Technique detected violation
- HTTP request component length checking failure
- Matching of a file type disallowed (or not allowed where required)
- Missing Mandatory header when required
- No match on allowed HTTP response code list
- Matching sensitive user data patterns as configured

c) SMTP security profile violations:
- SMTP protocol violation (RFC 2821 checks)
- Matching disallowed senders list
- DNS “A” record failure to resolve
- DNS SPF record validation failure
- SMTP matching disallowed methods as configured
- Activation of rate limiting/traffic shaping against a specific sender address

d) FTP security profile violations:
- Anonymous FTP requests
- Passive or active FTP requests
- FTP commands not in the allowed list
- Command line length exceeds the maximum length allowed
- FTP logon attempts exceed the maximum number allowed
- FTP traffic fails protocol compliance checks (RFC 959,1579,3659)]

6.2.1.8 FAU_STG.1 Protected audit trail storage

FAU_STG.1.1 The TSF shall protect the stored audit records in the audit trail from unauthorized deletion.

FAU_STG.1.2 The TSF shall be able to *prevent* unauthorized modifications to the stored audit records in the audit trail.

6.2.1.9 FAU_STG.3 Action in case of possible audit data loss

FAU_STG.3.1 The TSF shall [log a warning, delete the oldest audit records] if the audit trail exceeds [90% of the configured limit].

6.2.2 CRYPTOGRAPHIC SUPPORT (FCS)

6.2.2.1 FCS_CKM.1 (1) Cryptographic Key Generation – Asymmetric Keys

FCS_CKM.1.1(1) The TSF shall generate ***asymmetric*** cryptographic keys ***for Traffic encryption/decryption*** in accordance with a specified cryptographic key generation algorithm [hardware RNG] and specified cryptographic key sizes [1024 and 2048, 1024, 2048, and 4096**]** that meet the following: [ANSI X9.31**].**

6.2.2.2 FCS_CKM.1 (2) Cryptographic Key Generation –Symmetric Keys

FCS_CKM.1.1 (2) The TSF shall generate ***symmetric*** cryptographic keys ***for Traffic encryption/decryption*** in accordance with a specified cryptographic key generation algorithm [hardware RNG] and specified cryptographic key sizes [AES 128 and 256, AES 128 and 256] that meets the following [ANSI X9.31].

6.2.2.3 FCS_CKM.1 (3) Cryptographic Key Generation – PBKDF-generated Keys

FCS_CKM.1.1 (3) The TSF shall generate ***PBKDF-generated*** cryptographic keys in accordance with a specified cryptographic key generation algorithm [passphrase-based key derivation function (PBKDF)] and specified cryptographic key sizes [AES 192] that meets the following [PBKDF as implemented in the Linux 2.6 kernel according to RFC 2898].

6.2.2.4 FCS_COP.1 (1) Cryptographic operation – Secure Traffic

FCS_COP.1.1 (1) The TSF shall perform [Traffic Encryption/Decryption] in accordance with a specified cryptographic algorithm [per SSL ciphersuites listed in Appendix A] and cryptographic key sizes [as noted in Appendix A] that meet the following: [RFC 2246 (TLSv1.0), RFC3268 (AES – applies to both TLS and SSLv3), SSLv3 Specification (SSLv3)].

6.2.2.5 FCS_COP.1 (2) Cryptographic operation – Administrative-user Sessions

FCS_COP.1.1 (2) The TSF shall perform **[a**dministrative-user Session Encryption/Decryption] in accordance with a specified cryptographic algorithm [per SSL ciphersuites listed in Appendix A] and cryptographic key sizes [as noted in Appendix A] that meet the following: [RFC 2246 (TLSv1.0), RFC3268 (AES – applies to both TLS and SSLv3), SSLv3 Specification (SSLv3)].

6.2.2.6 FCS_COP.1 (3) Cryptographic operation – Cookie Encryption

FCS_COP.1.1 (3) The TSF shall perform [Cookie Encryption[5]] in accordance with a specified cryptographic algorithm [AES-CBC] and cryptographic key sizes [192 bits] that meet the following [RFC 3602].

6.2.2.7 FCS_COP.1 (4) Cryptographic operation - RSA signature/verification

FCS_COP.1.1 (4) The TSF shall perform [digital signature/verification of certificates**]** in accordance with a specified cryptographic algorithm [RSA**]** and cryptographic key sizes [1024, 2048, or 4096**]** that meet the following: [RFC 2246 (TLSv1.0), SSLv3 Specification (SSLv3)**].**

Note: The cryptography used in this product has not been FIPS certified nor has it been analyzed or tested to conform to cryptographic standards during this evaluation. All cryptography has only been asserted as tested by the vendor.

6.2.2.8 FCS_COP.1 (5) Cryptographic operation – Hashing

FCS_COP.1.1 (5) The TSF shall perform[hashing] in accordance with a specified cryptographic algorithm [MD5, SHA-1, RIPEMD-160] and ***~~cryptographic key~~ digest*** sizes [128, 160 bits] that meet the following: [RFC 1321, 3174, "The hash function RIPEMD-160"].

6.2.2.9 FCS_COP.1 (6) Cryptographic operation – SSH

FCS_COP.1.1 (6) The TSF shall perform [SSH Encryption/Decryption] in accordance with a specified cryptographic algorithm [per SSH ciphersuites listed in Appendix A] and cryptographic key sizes [as noted in Appendix A] that meet the following: [SSHv2 standard].

6.2.2.10 FCS_COP.1 (7) Cryptographic operation – Internal TSF data transfer

FCS_COP.1.1 (7) The TSF shall perform [internal TSF data transfer Encryption/Decryption] in accordance with a specified cryptographic algorithm [per SSL ciphersuites listed in Appendix A] and cryptographic key sizes [as noted in Appendix A] that meet the following: [RFC 2246 (TLSv1.0), RFC3268 (AES – applies to both TLS and SSLv3), SSLv3 Specification (SSLv3)].

6.2.2.11 FCS_COP.1 (8) Cryptographic operation – DSA signature/verification

FCS_COP.1.1 (8) The TSF shall perform [digital signature/verification of certificates**]** in accordance with a specified cryptographic algorithm [DSA **]** and cryptographic key sizes [1024**]** that meet the following: [FIPS 186-3**].**

Note: The cryptography used in this product has not been FIPS certified nor has it been analyzed or tested to conform to cryptographic standards during this evaluation. All cryptography has only been asserted as tested by the vendor.

[5] The cookie in this operation is simply AES encrypted as a string of data.

6.2.3 IDENTIFICATION AND AUTHENTICATION (FIA)

6.2.3.1 FIA_ATD.1 User attribute definition – Administrative-users

FIA_ATD.1.1 The TSF shall maintain the following list of security attributes belonging to individual ***Administrative***-users:

a) [identity;

b) association of a human user with a role defined in FMT_SMR.1;

c) Administrative Domain Partition Identifier[6]

d) Password Credentials]

6.2.3.2 FIA_UID.2 User identification before any action

FIA_UID.2.1 The TSF shall require each user to identify itself before allowing any other TSF-mediated actions on behalf of that user.

6.2.3.3 FIA_UAU.2 User authentication before any action

FIA_UAU.2.1 The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

6.2.3.4 FIA_UAU.5 Multiple authentication mechanisms

FIA_UAU.5.1 The TSF shall provide [Password, X.509 SSL Client/Server Certificate] to support user authentication.

FIA_UAU.5.2 The TSF shall authenticate any user's claimed identity according to the

- [Password: presentation of password credentials during Administrative-user GUI or tmsh session establishment that are successfully validated and associated with a valid account
- Client/Server SSL Certificate: presentation of a valid and non-revoked client/server certificate for authenticating SSL traffic between traffic user clients and the TOE and the TOE and backend servers associated with a valid account

]

6.2.3.5 FIA_AFL.1 Authentication failure handling

FIA_AFL.1.1 The TSF shall detect when *an administrator configurable positive integer within* [*1-65535,0 (0 means the feature is disabled)*[7]] unsuccessful authentication attempts occur related to [authorized TOE administrative-user

[6] The partition refers to the Administrative Domain partition on the BIG-IP. This allows an Administrative-user to define partitions and assign objects to that partition and then allow users to only see objects in a specific partition. It aids with information hiding.

[7] While 0-65535 are valid configurable values, the Common Criteria configuration requires that the Administrative-user set the number of authentication attempts to a non-zero value.

(with the exception of the default Administrative-user) access].

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been *met*, the TSF shall [disable the user account (with the exception of the default administrative-user account)].

6.2.3.6 FIA_SOS_EXP.1 Verification of secrets

FIA_SOS_EXP.1.1 The TSF shall provide a mechanism to verify that [*local passwords with the exception of those set by administrative-users with the Administrator or User Manager roles*] meet [*all of the following rules:*

- *requires at least 8 characters*
- *at least one character from each of the following: capital letters, lowercase letters, numbers, and punctuation;*
- *is not based on the userid or password entry,*
- *is not derived or derivable from the password entry;*
- *is not based on a dictionary word or reversed dictionary word, as defined by the systems dictionaries included with the Linux PAM module;*
- *does not match a former user password kept in password memory (configured as holding 0-127 former passwords per user) unless password memory is cleared by having an authorized administrative-user set the password;*
- *and has a minimum valid duration (configured as 0 days), a maximum valid duration (password expiration)(configured as 90 days), and a password expiration warning (configured as 7 days).*].

6.2.4 SECURITY MANAGEMENT (FMT)

6.2.4.1 FMT_SMF.1 Specification of Management Functions

FMT_SMF.1.1 The TSF shall be capable of performing the following security management functions: [

- PSM Security Profile Management
- Backup and Restore for TSF data, information flow rules (including iRules scripts and configuration data)
- Communication of authorized external IT entities with the TOE
- Enabling/Disabling of command audit functions[8]
- Startup/Shutdown TOE operations
- Information Flow Rules
- Proxy Translation Addresses configuration (SNAT)

[8] Note that the Common Criteria evaluated configuration recommends that audit logs remain enabled.

- Query Audit Logs
- Configure email alerts
- Virtual Server Management
- Administrative-user account policy management (including Password Policy, Authentication Failure configuration, and Session Timeout)
- Administrative-user account management (except changing own password)
- Administrative-user: Change own password
- View own user account information
- Node/Pool configuration
- Protocol profile configuration
- iRules script configuration
- Monitor
- Enable/Disable Nodes and Pool Members
- Authentication profile configuration
- SSL Profile Configuration
- Key and Certificate management
- Syslog server configuration
- OCSP server configuration
- TCP SYN flood attack mitigation
- Memory protection configuration
- Partition Management
- High availability configuration
- Profile configuration for profiles not explicitly mentioned above]

6.2.4.2 FMT_SAE.1 Time-limited authorization

FMT_SAE.1.1	The TSF shall restrict the capability to specify an expiration time for [user passwords] to [Administrator].
FMT_SAE.1.2	For each of these security attributes, the TSF shall be able to **[**invalidate the users authentication**]** after the expiration time for the indicated security attribute has passed.

6.2.4.3 FMT_SMR.1 Security roles

FMT_SMR.1.1	The TSF shall maintain the roles [Administrator, Manager, Application Editor, Operator, Guest, User Manager, Resource Administrator, Web Application Security Administrator, Web Application Security Editor, or No Access].
FMT_SMR.1.2	The TSF shall be able to associate users with roles.

6.2.4.4 FMT_MOF.1 Management of security functions behavior

FMT_MOF.1.1 The TSF shall restrict the ability to *determine the behavior of, modify the behavior of* the functions [listed in table Table 9: List of permissions by security function and role] to [roles listed in table Table 9: List of permissions by security function and role per the read (R) or (W) permissions listed].

	Administrator	Resource Administrator	Manager	User Manager	Application Editor	Operator	Guest	No Access	Web Application Security Administrator	Web Application Security Editor
PSM Security Profile Management	RW	R	R	R	R	R	R	--	R	R
Backup and Restore for TSF data, information flow rules (including iRules scripts and configuration data)	RW	RW	--	--	--	--	--	--	--	--
Communication of authorized external IT entities with the TOE	RW	R	R	R	R	R	R	--	R	R
Enabling/Disabling of command audit functions	RW	RW	--	--	--	--	--	--	--	--
Startup/Shutdown TOE operations	RW	RW	--	--	--	--	--	--	--	--
Information Flow Rules	RW	RW	R	R	R	R	R	--	R	R
Proxy Translation Addresses configuration (SNAT)	RW	RW	R	R	R	R	R	--	R	R
Query Audit Logs	RW	RW	--	--	--	--	--	--	--	--
Configure email alerts	RW	--	--	--	--	--	--	--	--	--
Virtual Server Management	RW	RW	RW	R	R	R	R	--	R	R
Administrative-user account policy management (including Password Policy, Authentication Failure configuration, and Session Timeout)	RW	R	R	R	R	R	R	--	R	R

	Administrator	Resource Administrator	Manager	User Manager	Application Editor	Operator	Guest	No Access	Web Application Security Administrator	Web Application Security Editor
Administrative-user account management (except changing own password)	RW	R	--	RW[9]			--	--		
Administrative-user: Change own password	RW	RW	RW	RW	RW	RW	RW	--	RW	RW
View own user account information	RW	R	R	RW	R	R	R	--	R	R
Node/Pool configuration	RW	RW	RW	R	RW[10]	R	R	--	R	R
Protocol profile configuration	RW	RW	RW	R	R	R	R	--	R[11]	R[12]
iRules script configuration	RW	RW	RW	R	R	R	R	--	R	R
Monitor	RW	RW	RW	R	R	R	R	--	R	R
Enable/Disable Nodes and Pool Members	RW	RW	RW	R	RW	RW	R	--	R	R
Authentication profile configuration	RW	RW	RW	R	R	R	R	--	R	R
SSL Profile Configuration	RW	RW	RW	R	R	R	R	--	R	R
Key and Certificate management	RW	RW	R	R	R	R	R	--	R	R
Syslog server configuration	RW	RW	R	R	R	R	R	--	R	R
OCSP server configuration	RW	RW	RW	R	R	R	R	--	R	R
TCP SYN flood attack mitigation	RW	RW	R	R	R	R	R	--	R	R

[9] Except for users with the administrator role, which are read-only..

[10] Can modify nodes and pools but not create or delete them

[11] RW for HTTP class protocol profiles

[12] RW for HTTP class protocol profiles

	Administrator	Resource Administrator	Manager	User Manager	Application Editor	Operator	Guest	No Access	Web Application Security Administrator	Web Application Security Editor
Memory protection configuration	RW	RW	R	R	R	R	R	--	R	R
Partition Management	RW	RW	R	R	R	R	R	--	R	R
High Availability Configuration	RW	RW	R	R	R	R	R	-	R	R
Profile configurations for profiles not explicitly mentioned above	RW	RW	RW	R	R	R	R	-	R	R

Table 9: List of permissions by security function and role

6.2.4.5 FMT_MSA.1 Management of security attributes (1)

FMT_MSA.1.1 (1) The TSF shall enforce the [UNAUTHENTICATED SFP] to restrict the ability to [*delete attributes from a rule, modify attributes in a rule, add attributes to a rule*] the security attributes [listed in section FDP_IFF.1.1(1)] to [Administrator, Resource Administrator, Manager].

6.2.4.6 FMT_MSA.1 Management of security attributes (2)

FMT_MSA.1.1 (2) The TSF shall enforce the [AUTHENTICATED SFP] to restrict the ability to [*delete attributes from a rule, modify attributes in a rule, add attributes to a rule*] the security attributes [information flow rules described in FDP_IFF.1 (2)] to [Administrator, Resource Administrator, Manager].

6.2.4.7 FMT_MSA.3 Static attribute initialization

FMT_MSA.3.1 (1) The TSF shall enforce the [UNAUTHENTICATED_SFP and AUTHENTICATED_SFP] to provide *restrictive* default values for ***information flow (with the exception of IP address and port)*** security attributes that are used to enforce the SFP.

FMT_MSA.3.1 (2) The TSF shall enforce the [UNAUTHENTICATED_SFP and AUTHENTICATED_SFP] to provide *restrictive* default values for ***information flow (IP address and port only, configured through the GUI)*** security attributes that are used to enforce the SFP.

FMT_MSA.3.1 (3) The TSF shall enforce the [UNAUTHENTICATED_SFP and AUTHENTICATED_SFP] to provide *permissive* default values for

information flow (IP address and port only, configured through tmsh) security attributes that are used to enforce the SFP.

FMT_MSA.3.2 (1) The TSF shall allow [Administrator, Resource Administrator] to specify alternative initial values to override the default values when an object or information is created.

FMT_MSA.3.2 (2) The TSF shall allow [Administrator, Resource Administrator] to specify alternative initial values to override the default values when an object or information is created.

FMT_MSA.3.2 (3) The TSF shall allow [Administrator, Resource Administrator] to specify alternative initial values to override the default values when an object or information is created.

6.2.4.8 FMT_MTD.1 Management of TSF data (1)

FMT_MTD.1.1 (1) The TSF shall restrict the ability to *query, modify, delete,* [*create and assign*] the [user attributes defined in FIA_ATD.1.1 except for the user's own password (except for the Administrator role users] to [Administrator, User Manager].

6.2.4.9 FMT_MTD.1 Management of TSF data (2)

FMT_MTD.1.1 (2) The TSF shall restrict the ability to [*set*] the [all TSF data except as noted in FMT_MTD.1(1), FMT_MTD.1(3), FMT_MTD.1(4), FMT_MTD.1(5), FMT_MTD.1(6), FMT_MTD.1(7)] to [Administrator, Resource Administrator].

6.2.4.10 FMT_MTD.1 Management of TSF data (3)

FMT_MTD.1.1 (3) The TSF shall restrict the ability to *delete,* [*create and assign*] the [Administrative-user's (with the exception of the Administrator role) own password] to [Administrator, User Manager].

6.2.4.11 FMT_MTD.1 Management of TSF data (4)

FMT_MTD.1.1 (4) The TSF shall restrict the ability to *modify* the [Administrative-user's (with the exception of the Administrator role) own password] to [the owner of that password, Administrator, User Manager.]

6.2.4.12 FMT_MTD.1 Management of TSF data (5)

FMT_MTD.1.1 (5) The TSF shall restrict the ability to *delete,* [*create and assign*] the [Administrator role user's own password] to [Administrator].

6.2.4.13 FMT_MTD.1 Management of TSF data (6)

FMT_MTD.1.1 (6) The TSF shall restrict the ability to *modify* the [Administrator role user's own password] to [the owner of that password, Administrator.]

6.2.4.14 FMT_MTD.1 Management of TSF data (7)

FMT_MTD.1.1 (7) The TSF shall restrict the ability to *query, modify, delete,* [*create and assign*] the [user attributes defined in FIA_ATD.1.1 except for the user's own password (for the Administrator role)] to [Administrator].

6.2.5 PROTECTION OF THE TSF (FPT)

6.2.5.1 FPT_ FLS.1 Failure with preservation of secure state

FPT_FLS.1.1 The TSF shall preserve a secure state when the following types of failures occur: [

- Operational failure of a single TOE hardware device]

6.2.5.2 FPT_FLS_EXP.1 Failure with preservation of secure configuration state

FPT_FLS_EXP.1.1 The TSF shall provide [Administrative-users] an interface to preserve a secure configuration state.

6.2.5.3 FPT_ITC.1 Inter-TSF confidentiality during transmission

FPT_ITC.1.1 The TSF shall protect all TSF data transmitted from the TSF to a remote trusted IT product[13] from unauthorized disclosure during transmission.

6.2.5.4 FPT_ITI.1 Inter-TSF detection of modification

FPT_ITI.1.1 The TSF shall provide the capability to detect modification of all TSF data during transmission between the TSF and a remote trusted IT product[14] within the following metric: [a single Message Authentication Code (MAC) error during transmission**].**

FPT_ITI.1.2 The TSF shall provide the capability to verify the integrity of all TSF data transmitted between the TSF and a remote trusted IT product[15] and perform [resending of transmitted data**]** if modifications are detected.

6.2.5.5 FPT_ITT.1 Basic internal TSF data transfer protection

FPT_ITT.1.1 The TSF shall protect TSF data from [*disclosure, modification*] when it is transmitted between separate parts of the TOE.

6.2.5.6 FPT_STM.1 Reliable time stamps

FPT_STM.1.1 The TSF shall be able to provide reliable time stamps.

6.2.6 USER DATA PROTECTION: INFORMATION FLOW CONTROL

---

[13] Refers to sessions with an external authentication server in the Operational Environment

[14] Refers to sessions with an external authentication server in the Operational Environment

[15] Refers to sessions with an external authentication server in the Operational Environment

(FDP)

6.2.6.1 FDP_IFC.1 (1) Subset information flow control - UNAUTHENTICATED

FDP_IFC.1.1 (1) The TSF shall enforce the [UNAUTHENTICATED SFP] on:

a) [subjects: unauthenticated external IT entities that send and receive information through the TOE to one another;

b) information: traffic sent through the TOE from one subject to another;

c) operation: pass information;

d) operation: reject information].

6.2.6.2 FDP_IFC.1(2) Subset information flow control - AUTHENTICATED

FDP_IFC.1.1 (2) The TSF shall enforce the [AUTHENTICATED SFP] on

a) [subjects: authenticated external IT entities that send and receive information through the TOE to one another;

b) information: traffic sent through the TOE from one subject to another;

c) operation: pass information;

d) operation: reject information].

6.2.6.3 FDP_IFF.1 (1) Simple security attributes - UNAUTHENTICATED

FDP_IFF.1.1 (1) The TSF shall enforce the [UNAUTHENTICATED SFP] based on the following types of subject and information security attributes:

a) [subject security attributes:

- presumed address;

b) information security attributes:

- presumed address of source subject;
- presumed address of destination subject;
- transport layer protocol;
- TOE interface on which traffic arrives and departs;
- TOE service which processes traffic;
- HTTP protocol characteristics (parameter lengths);
- HTTP coding methods (evasion techniques; disallowed list);
- HTTP Request Component Length (i.e.: “URI”, “Query String”, “Post Data”, “Full Request”);
- HTTP “File Type” association (if configured);
- HTTP “Mandatory Header” association (if configured);
- HTTP “Response Codes List” association (if configured);
- HTTP User Data patterns (data type) associated with sensitive data
- SMTP Protocol characteristics (parameter lengths)

- FTP protocol characteristics (command/arg. length)
- FTP request characteristics
- FTP mode
- FTP user name]

FDP_IFF.1.2(1) The TSF shall permit an information flow between a controlled subject and ***another*** controlled ~~***information***~~ ***subject*** via a controlled operation if the following rules hold:

a) [Subjects on an internal network can cause information to flow through the TOE to another connected network if all of the following are met:

- all the information security attribute values are unambiguously permitted by the information flow security policy rules, where such rules may be composed from all possible combinations of the values of the information flow security attributes, created by the authorized administrative-user;
- the presumed address of the destination subject, in the information, matches an address configured on a virtual server that is enabled on that VLAN, SNAT, or NAT object on the BIG-IP;
- and enabled iRules permit the information flow.

b) Subjects on the external network can cause information to flow through the TOE to another connected network if all of the following are met:

- all the information security attribute values are unambiguously permitted by the information flow security policy rules, where such rules may be composed from all possible combinations of the values of the information flow security attributes, created by the authorized administrative-user;
- the presumed address of the destination subject, in the information,matches an address configured on a virtual server, SNAT, or NAT object on the BIG-IP;
- and enabled iRules permit the information flow.]

FDP_IFF.1.3 (1) The TSF shall enforce the [no additional information flow control rules].

FDP_IFF.1.4 (1) The TSF shall explicitly authorize an information flow based on the following rules: [Traffic: Traffic is explicitly authorized within the same session based on the previous session authentication for the time specified in the configuration by the persistence data and connection table timeouts.]

FDP_IFF.1.5 (1) The TSF shall explicitly deny an information flow based on the following rules:

a) [Using the Reaper High Water Mark function, the TOE will stop accepting new connections based on Administrative-user configured memory usage settings to avoid a Denial of Service type attack.

b) Packets that are determined to be malformed or do not meet protocol standards are rejected and discarded to protect TOE resources. The TOE shall reject requests for access or services where the information arrives on an external TOE interface, and the presumed address of the source subject is an external IT entity on an internal network;

c) The TOE shall reject requests in which the subject specifies the route in which information shall flow en route to the receiving subject;

d) The TOE shall reject requests if so specified by an enabled iRule.]

6.2.6.4 FDP_IFF.1 (2) Simple security attributes - AUTHENTICATED

FDP_IFF.1.1 (2) The TSF shall enforce the [AUTHENTICATED SFP] based on the following types of subject and information security attributes**:** [

a. Subject Security Attributes:

- presumed address
- username, password/X.509 Certificate

b. Information Security Attributes:

- presumed address of source subject;
- presumed address of destination subject;
- transport layer protocol;
- X.509 certificate expiration date
- X.509 certificate revocation status
- TOE interface on which traffic arrives and departs;
- TOE service which processes traffic;
- HTTP protocol characteristics (parameter lengths)
- HTTP coding methods (evasion techniques; disallowed list)
- HTTP Request Component Length (i.e.: “URI”, “Query String”, “Post Data”, “Full Request”)
- HTTP “File Type” association (if configured)
- HTTP “Mandatory Header” association (if configured)
- HTTP “Response Codes List” association (if configured)
- HTTP User Data patterns (data type) associated with sensitive data
-]

FDP_IFF.1.2 (2) The TSF shall permit an information flow between a controlled subject and ***another*** controlled ~~***information***~~ ***subject*** via a controlled operation if the following rules hold: [

a. [Subjects on an internal network can cause information to flow through

the TOE to another connected network if all of the following are met:

- Successful negotiation of SSL protocol; required key exchange has successfully taken place, certificate verification and revocation checks are successful; username/password successfully authenticates,;
- all the information security attribute values are unambiguously permitted by the information flow security policy rules, where such rules may be composed from all possible combinations of the values of the information flow security attributes, created by the authorized administrative-user;
- the presumed address of the destination subject, in the information,matches an address configured on a virtual server that is enabled on that VLAN, SNAT, or NAT object on the BIG-IP;
- and enabled iRules permit the information flow.

b. Subjects on the external network can cause information to flow through the TOE to another connected network if all of the following are met:

- Successful negotiation of SSL protocol; username/password combination resolves to a valid authenticated role, required key exchange has successfully taken place, certificate verification and revocation checks are successful;
- all the information security attribute values are unambiguously permitted by the information flow security policy rules, where such rules may be composed from all possible combinations of the values of the information flow security attributes, created by the authorized administrative-user;
- the presumed address of the destination subject, in the information,matches an address configured on a virtual server, SNAT, or NAT object on the BIG-IP
- and enabled iRules permit the information flow.]

FDP_IFF.1.3 (2) The TSF shall enforce the: [no additional information flow control rules.]

FDP_IFF.1.4 (2) The TSF shall explicitly authorize an information flow based on the following rules: [Traffic: Traffic is explicitly authorized within the same session based on the previous session authentication for the time specified in the configuration by the persistence data and connection table timeouts.]]

FDP_IFF.1.5 (2) The TSF shall explicitly deny an information flow based on the following rules:

a. [Using the Reaper High Water Mark function, the TOE will stop accepting new connections based on Administrative-user configured memory usage settings to avoid a Denial of Service type attack.

b. Packets that are determined to be malformed or do not meet protocol standards are rejected and discarded to protect TOE resources.

c. The TOE shall reject requests in which the subject specifies the route in which information shall flow en route to the receiving subject;

d. The TOE shall reject requests if so specified by an enabled iRule.]

6.2.6.5 FDP_RIP.1 Subset residual information protection

FDP_RIP.1.1 The TSF shall ensure that any previous information content of a resource is made unavailable upon the *allocation of the resource to* [all objects].

6.2.7 TOE Access (FTA)

6.2.7.1 FTA_SSL_EXP.3 TSF Initiated Termination

FTA_SSL_EXP.3.1 The TSF shall terminate a [GUI] interactive session after an [Administrator, User Manager configured value (default 20 minutes)].

6.2.7.2 FTA_SSL.4 User-initiated termination

FTA_SSL.4.1 The TSF shall allow user-initiated termination of the user's own interactive session.

6.2.8 Resource Allocation (FRU)

6.2.8.1 FRU_FLT.1 Degraded fault tolerance

FRU_FLT.1.1 The TSF shall ensure the operation of [maintain full TOE functionality] when the following failures occur: [any failure of a single appliance].

6.2.8.2 FRU_PRS.1 Limited priority of service

FRU_PRS.1.1 The TSF shall assign a priority to each subject in the TSF.

FRU_PRS.1.2 The TSF shall ensure that each access to [TOE backend server] shall be mediated on the basis of the subject's assigned priority.

6.2.8.3 FRU_RSA_EXP.1(1) Maximum quotas – *TCP Connections*

FRU_RSA_EXP.1.1 (1) The TSF shall enforce maximum quotas of the following resources: [TCP connections] that *the TOE* can have active *simultaneously***.**

6.2.8.4 FRU_RSA_EXP.1(2) Maximum quotas – TCP SYN queue entries

FRU_RSA_EXP.1.1 (2) The TSF shall enforce maximum quotas of the following resources: [TCP SYN queue entries] that *the TOE* can have active *simultaneously***.**

6.2.9 Trusted path/channels (FTP)

6.2.9.1 FTP_ITC.1 Inter-TSF trusted channel

FTP_ITC.1.1 The TSF shall provide a communication channel between itself and another trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.

FTP_ITC.1.2 The TSF shall permit *another trusted IT product* to initiate communication via the trusted channel.

FTP_ITC.1.3 The TSF shall initiate communication via the trusted channel for [functions listed in FMT_MOF.1].

6.2.10 Security Management - Explicit

6.2.10.1 FMT_SCR_EXP.1 Scripting of Custom Flow Control Rules

FMT_SCR_EXP.1.1 The TSF shall support the development of custom flow control rules using a scripting language for use in constructing a rules set.

FMT_SCR_EXP.1.2 The TSF shall provide a Scripting language which includes the following constructs at a minimum: *Event Declarations, Operators, Commands* [none] with syntax based on the [Tool Command Language (TCL)] standard.

FMT_SCR_EXP.1.3 The TSF shall support the custom flow control rules scripted in FMT_SCR_EXP.1.1 to be used in augmenting the following Flow Control Security Functional Policy(s): [UNAUTHENTICATED SFP, AUTHENTICATED SFP]

FMT_SCR_EXP.1.4 The TSF shall implement the scripted action in the event the conditional rule set statements are TRUE and shall not implement the scripted action in the event the conditional rule set statements are FALSE based on the [UNAUTHENTICATED SFP, AUTHENTICATED SFP] defined in FDP_IFF.1***(1), (2)***.

FMT_SCR_EXP.1.5 The TSF shall support implementation of the following functions using the scripting language command set: [

- Selecting traffic destination based on scripted parameter
- Query header or content and return data
- Perform data manipulation including insertion of headers into HTTP requests
- Parse and manipulate content including decoding and return result
- Perform a search on packet headers using scripted parameters, perform analysis based on scripted parameters and direct packets based on configured parameters
- Perform a search of packet contents using scripted parameters perform analysis based on scripted parameters and direct packets based on configured parameters
- Direct traffic to configured Pools, Individual pools members, specific ports, URI paths in order to implement persistence settings or load balancing objectives
- Detecting sensitive data patterns and inserting null data into the data stream to sanitize sensitive data flows (SSN from 123-45-6789 to xxx-xx-xxxx)]

6.2.11 User Data Protection - Explicit

6.2.11.1 FDP_PXY_EXP.1 Reverse Proxy

FDP_PXY_EXP.1.1 The TSF shall rewrite [UDP/IP and TCP/IP packet headers (including checksum)] for [all UDP/IP, TCP/IP] packets traversing the TOE between internal and external networks.

FDP_PXY_EXP.1.2 The TSF shall rewrite Inbound [UDP /IP and TCP/IP packet headers] from external addresses to internal addresses as configured by the [Administrator, Resource Administrator, Manager].

FDP_PXY_EXP.1.3 The TSF shall rewrite Outbound [UDP/IP and TCP/IP packet headers] from internal to external IP Addresses as configured by the [Administrator, Resource Administrator, Manager].

FDP_PXY_EXP.1.4 The TSF shall checksum packet headers by generating and verifying a [16 bit] CRC per [IPv4, IPv6].

FDP_PXY_EXP.1.5 The TSF shall create discrete buffers for each packet, associate the buffers to a data flow and process each flow separately.

6.3 Rationale for TOE Security Requirements

	O.IDAUTH	O.MEDIAT	O.SECSTA	O.CRYPTO	O.SELPRO	O.AUDREC	O.AUDALERT	O.ACCOUN	O.SECFUN	O.LIMEXT	O.RESOURCE_X	O.SESSION_TERM	O.SAFE_FAIL	O.PRIORITY	O.SCRIPT_FUNC	O.PROXY
FMT_SMF.1									X							
FMT_SMR.1									X							
FIA_ATD.1	X								X							
FIA_UID.2	X							X								
FIA_SOS_EXP.1	X															
FIA_AFL.1					X											
FIA_UAU.2	X							X								
FIA_UAU.5	X							X								
FDP_IFC.1 (1)		X														
FDP_IFF.1 (1)		X														
FDP_IFC.1 (2)		X														
FDP_IFF.1 (2)		X														
FMT_MSA.1 (1)		X	X						X							
FMT_MSA.1 (2)		X	X						X							
FMT_MSA.3 (1)		X	X													
FMT_MSA.3 (2)		X	X													
FMT_MSA.3 (3)		X	X													
FMT_MTD.1 (1)									X							

	O.IDAUTH	O.MEDIAT	O.SECSTA	O.CRYPTO	O.SELPRO	O.AUDREC	O.AUDALERT	O.ACCOUN	O.SECFUN	O.LIMEXT	O.RESOURCE_X	O.SESSION_TERM	O.SAFE_FAIL	O.PRIORITY	O.SCRIPT_FUNC	O.PROXY
FMT_MTD.1 (2)									X							
FMT_MTD.1 (3)									X							
FMT_MTD.1 (4)									X							
FMT_MTD.1 (5)									X							
FMT_MTD.1 (6)									X							
FMT_MTD.1 (7)									X							
FMT_SAE.1									X							
FDP_RIP.1		X														
FCS_CKM.1 (1)				X												
FCS_CKM.1 (2)				X												
FCS_CKM.1 (3)				X												
FCS_COP.1 (1)				X												
FCS_COP.1 (2)				X												
FCS_COP.1 (3)				X												
FCS_COP.1 (4)				X												
FCS_COP.1 (5)				X												
FCS_COP.1 (6)				X												
FCS_COP.1 (7)				X												
FCS_COP.1 (8)				X												
FPT_STM.1						X										
FPT_FLS.1													X			
FPT_FLS_EXP.1													X			
FPT_ITC.1					X											
FPT_ITI.1																
FPT_ITT.1																
FAU_ARP.1							X				X					
FAU_GEN.1						X		X								
FAU_GEN.2						X		X								
FAU_SAA.1							X				X					
FAU_SAA_EXP.1							X				X					
FAU_SAR.1						X										
FAU_SAR.3						X										
FAU_STG.1			X		X				X							
FAU_STG.3			X		X				X							
FMT_MOF.1			X							X						
FTA_SSL_EXP.3												X				
FTA_SSL.4												X				
FRU_FLT.1													X			

	O.IDAUTH	O.MEDIAT	O.SECSTA	O.CRYPTO	O.SELPRO	O.AUDREC	O.AUDALERT	O.ACCOUN	O.SECFUN	O.LIMEXT	O.RESOURCE_X	O.SESSION_TERM	O.SAFE_FAIL	O.PRIORITY	O.SCRIPT_FUNC	O.PROXY
FRU_PRS.1														X		
FRU_RSA_EXP.1(1)											X					
FRU_RSA_EXP.1(2)											X					
FTP_ITC.1									X							
FMT_SCR_EXP.1															X	
FDP_PXY_EXP.1																X

Table 10: Summary of Mappings between Security Functions and IT Security Objectives

6.3.1 TOE Security Functional Requirements Rationale

The security requirements are derived according to the general model presented in Part 1 of the Common Criteria. Specifically, Table 5 illustrates the mapping between the security requirements and the security objectives and Table 4: Summary of Mappings between Threats and IT Security demonstrates the relationship between the threats, policies and IT security objectives. The functional and assurance requirements presented in this Security Target are mutually supportive and their combination meets the stated security objectives.

FMT_SAE.1 Time-limited authorization

FMT_SAE.1 ensures that the Administrative-user has the ability to specify an expiration time for passwords and that the TSF has the ability to invalidate authentication upon expiration of passwords. This component traces back to and aids in meeting the following objective: O.SECFUN.

FMT_SMF.1 Specification of Management Functions

This component describes the Security Management functions that may be invoked through the Administrative-user interface for management of the TOE. This component traces back to and aids in meeting the following objective: O.SECFUN.

FMT_SMR.1 Security roles

This component requires the TSF to support Security Roles for TOE users. This component traces back to and aids in meeting the following objective: O.SECFUN.

FIA_ATD.1 Administrative-user attribute definition

This component exists to provide users with attributes to distinguish one administrative-user from another, for accountability purposes and to associate the role chosen in FMT_SMR.1 with a user. This component traces back to and aids in meeting the following objectives: O.IDAUTH and O.SECFUN.

FIA_UID.2 User identification before any action

This component ensures that before anything occurs on behalf of a user, the user's identity is identified to the TOE. This component traces back to and aids in meeting the following objectives: O.IDAUTH and O.ACCOUN.

FIA_SOS_EXP.1 Verification of Secrets

This component ensures that passwords used for Administrative-users meet the minimum password complexity policy enforced by the TSF. This component traces back to and aids in meeting the following objectives: O.IDAUTH.

FIA_AFL.1 Authentication failure handling

This component ensures that human users who are not authorized administrative-users can not endlessly attempt to authenticate. After the configured number of failed login attempts (more than zero) the user's attempts to authenticate fail. This situation remains in effect until an authorized administrative-user makes authentication possible again for that user. This component traces back to and aids in meeting the following objective: O.SELPRO.

FIA_UAU.2 User authentication before any action

This component ensures that before anything occurs on behalf of a user, the user must authenticate to the TOE. This component traces back to and aids in meeting the following objectives: O.IDAUTH and O.ACCOUN.

FIA_UAU.5 Multiple authentication mechanisms

This component provides multiple methods of authentication based on the purpose. Password and X.509 certificate based authentication is supported by the TOE. This component traces back to and aids in meeting the following objectives: O.IDAUTH and O.ACCOUN.

FDP_IFC.1(1) Subset information flow control (1)

This component identifies the entities involved in the UNAUTHENTICATED information flow control SFP (i.e., users sending information to other users and vice versa). This component traces back to and aids in meeting the following objective: O.MEDIAT.

FDP_IFC.1(1) Subset information flow control (2)

This component identifies the entities involved in the AUTHENTICATED SFP. This component traces back to and aids in meeting the following objective: O.MEDIAT.

FDP_IFF.1(1) Simple security attributes (1)

This component identifies the attributes of the users sending and receiving the information in the UNAUTHENTICATED SFP, as well as the attributes for the information itself. Then the policy is defined by stating under what conditions information is permitted to flow. This component traces back to and aids in meeting the following objective: O.MEDIAT.

FDP_IFF.1(2) Simple security attributes (2)

This component identifies the attributes of authenticated users sending and receiving the information to backend server as per the AUTHENTICATED SFP, as well as the attributes for the information itself. Then the policy is defined by stating under what conditions information is permitted to flow from resources requiring authentication. This component traces back to and aids in meeting the following objective: O.MEDIAT.

FMT_MSA.1(1) Management of security attributes (1)

This component ensures the TSF enforces the AUTHENTICATED_SFP to restrict the ability to delete, modify, and add within a rule those security attributes that are listed in section FDP_IFF1.1(1). This component traces back to and aids in meeting the following objectives: O.MEDIAT, O.SECSTA, and O.SECFUN.

FMT_MSA.1(2) Management of security attributes (2)

This component ensures the TSF enforces the AUTHENTICATED_SFP to restrict the ability to create or delete rules for security attributes that are listed in FDP_IFF.1(1). This component traces back to and aids in meeting the following objectives: O.MEDIAT, O.SECSTA, and O.SECFUN.

FMT_MSA.3 (1) Static attribute initialization

This component ensures that there is a default deny policy for the information flow control security rules for all security attributes except for IP address and port. This component traces back to and aids in meeting the following objectives: O.MEDIAT and O.SECSTA.

FMT_MSA.3 (2) Static attribute initialization

This component ensures that there is a default deny policy for the information flow control security rules for IP address and port configured through the GUI. This component traces back to and aids in meeting the following objectives: O.MEDIAT and O.SECSTA.

FMT_MSA.3 (3) Static attribute initialization

This component ensures that there is a default allow policy for the information flow control security rules for IP address and port configured through tmsh. This component traces back to and aids in meeting the following objectives: O.MEDIAT and O.SECSTA.

FMT_MTD.1(1) Management of TSF data (1)

This component ensures that the TSF restrict abilities to query, modify, delete and assign certain Administrative-user attributes as defined in FIA_ATD.1.1 (with the exception of the user's own password) to only the authorized Administrative-user. This component traces back to and aids in meeting the following objective: O.SECFUN.

FMT_MTD.1 (2) Management of TSF data (2)

This component ensures that the TSF restricts abilities to set the time and date used to form timestamps to only the authorized Administrative-user. This component traces back to and aids in meeting the following objective: O.SECFUN.

FMT_MTD.1 (3) Management of TSF data (3)

This component ensures that the TSF restrict abilities to query, delete and assign an Administrative user's own password to only the authorized Administrative-user. This component traces back to and aids in meeting the following objective: O.SECFUN

FMT_MTD.1 (4) Management of TSF data (4)

This component ensures that the TSF restrict abilities to modify an Administrative user's own password to that user, and an authorized Administrative-user. This component traces back to and aids in meeting the following objective: O.SECFUN

FMT_MTD.1 (5) Management of TSF data (5)

This component ensures that the TSF restrict abilities to query, delete and assign an Administrative user's own password to only the authorized Administrative-user. This component traces back to and aids in meeting the following objective: O.SECFUN

FMT_MTD.1 (6) Management of TSF data (6)

This component ensures that the TSF restrict abilities to modify an Administrative user's own password to that user, and an authorized Administrative-user. This component traces back to and aids in meeting the following objective: O.SECFUN

FMT_MTD.1(7) Management of TSF data (7)

This component ensures that the TSF restrict abilities to query, modify, delete and assign certain Administrative-user attributes as defined in FIA_ATD.1.1 (with the exception of the user's own password) to only the authorized Administrative-user. This component traces back to and aids in meeting the following objective: O.SECFUN.

FPT_FLS.1 Failure with preservation of Secure State

This component ensures that the TSF maintains a secure state following the specified failure conditions and aids in meeting the following objective: O.SAFE_FAIL.

FPT_FLS_EXP.1 Failure with preservation of Secure Configuration State

This component ensures that the TSF provides an interface to maintain a secure configuration state and aids in meeting the following objective: O.SAFE_FAIL.

FPT_ITC.1 Inter-TSF confidentiality during transmission

This component ensures that the TSF assures confidentiality of transmissions made to trusted IT resources in the Operational Environment i.e.: authentication servers.

FPT_ITI.1 Inter-TSF detection of modification

This component ensure that if modifications occur during communication with trusted IT resources in the Operational Environment that they are detected and the specified action is taken (resending) upon a single MAC error.

FPT_ITT.1 Basic internal TSF data transfer protection

This component ensures that TSF data is protected when transferred between separate parts of the TOE across an internal channel.

FDP_RIP.1 Subset residual information protection

This component ensures that neither information that had flowed through the TOE nor any TOE internal data are used when padding is used by the TOE for information flows. This component traces back to and aids in meeting the following objective: O.MEDIAT.

FCS_CKM.1(1) Cryptographic Key Generation – Asymmetric Keys

This component ensures that the specified key generation algorithm is used for generating asymmetric cryptographic keys used as part of performing DH key exchange for secure sessions and traces to the following objective: O.CRYPTO.

FCS_CKM.1(2) Cryptographic Key Generation – Symmetric Keys

This component ensures that the specified key generation algorithm is used for generating symmetric cryptographic keys and traces to the following objective: O.CRYPTO.

FCS_CKM.1(3) Cryptographic Key Generation – PBKDF-generated Keys

This component ensures that the specified key generation algorithm is used for generating Passphrase-Based Key Definition Function cryptographic keys and traces to the following objective: O.CRYPTO.

FCS_COP.1 (1) Cryptographic operation – Secure Traffic

This component ensures that specified ciphersuites are used to encrypt SSL sessions for secure traffic mediated by the TOE in accordance with the specified standard. This component traces back to and aids in meeting the following objective: O.CRYPTO.

FCS_COP.1 (2) Cryptographic operation – Administrative-user Sessions

This component ensures that specified algorithms are used to encrypt SSL sessions for Administrative-user sessions with the TOE in accordance with the specified standard. This component traces back to and aids in meeting the following objective: O.CRYPTO.

FCS_COP.1 (3) Cryptographic operation – Cookie Encryption

This component ensures that the specified algorithms and key sizes are used to encrypt/decrypt cookies passed between the BIG-IP appliance and Client users. This component traces back to and aids in meeting the following objective: O.CRYPTO .

FCS_COP.1 (4) Cryptographic operation – Digital Signature/Verification (RSA)

This component ensures that the specified algorithms and key sizes are used to sign/verify digital signatures for self signed certificates. This component traces back to and aids in meeting the following objective: O.CRYPTO.

FCS_COP.1 (5) Cryptographic operation - Hashing

This component ensures that the specified algorithms and digest sizes are used to perform hashing. This component traces back to and aids in meeting the following objective: O.CRYPTO.

FCS_COP.1 (6) Cryptographic operation – SSH

This component ensures that specified ciphersuites are used to encrypt SSH sessions for secure traffic mediated by the TOE in accordance with the specified standard. This component traces back to and aids in meeting the following objective: O.CRYPTO.

FCS_COP.1 (7) Cryptographic operation – Internal TSF data transfer

This component ensures that specified algorithms are used to encrypt SSL sessions for internal data transfer within the TOE in accordance with the specified standard. This component traces back to and aids in meeting the following objective: O.CRYPTO.

FCS_COP.1 (8) Cryptographic operation – Digital Signature/Verification (DSA)

This component ensures that the specified algorithms and key sizes are used to sign/verify digital signatures for self signed certificates. This component traces back to and aids in meeting the following objective: O.CRYPTO.

FPT_STM.1 Reliable time stamps

FAU_GEN.1 depends on this component. It ensures that the date and time on the TOE is dependable. This is important for the audit trail. This component traces back to and aids in meeting the following objective: O.AUDREC.

FAU_ARP.1 Security Alarms

This component assures that the TOE takes the specified action in the event a security violation is detected in accordance with criteria named in SFRs FAU_SAA.1, FAU_SAA_EXP.1. This component traces back to and aids in meeting the following objective: O.MEDIATE, O.RESOURCE_X.

FAU_GEN.1 Audit data generation

This component outlines what data must be included in audit records and what events must be audited. This component traces back to and aids in meeting the following objectives: O.AUDREC and O.ACCOUN.

FAU_GEN.2 User Identity Association

This component specifies that audit records must be associated with the user identity that initiated the logged event. This component traces back to and aids in meeting the following objectives: O.AUDREC and O.ACCOUN.

FAU_SAA.1 Potential Violation Analysis

This component assures that the TOE performs analysis based on configured criteria to determine when a security violation is detected and takes the specified action as name in SFR FAU_ARP.1. This component traces back to and aids in meeting the following objective: O.MEDIATE, O.RESOURCE_X.

FAU_SAA_EXP.1 Potential Violation Analysis

This component assures that the TOE performs analysis based on configured criteria to determine when a security violation is detected and takes the specified action as name in SFR FAU_ARP.1. This component traces back to and aids in meeting the following objective: O.MEDIATE, O.RESOURCE_X.

FAU_SAR.1 Audit review

This component ensures that the audit trail is understandable. This component traces back to and aids in meeting the following objective: O.AUDREC.

FAU_SAR.3 Selectable audit review

This component ensures that a variety of searches and sorts can be performed on the audit trail. This component traces back to and aids in meeting the following objective: O.AUDREC.

FAU_STG.1 Protected audit trail storage

This component is chosen to ensure that the audit trail is protected from tampering, the security functionality is limited to the authorized Administrative-user, and that start-up and recovery does not compromise the audit records. This component traces back to and aids in meeting the following objectives: O.SELPRO, O.SECFUN, and O.SECSTA.

FAU_STG.3 Action in case of possible audit data loss

This component defines how the TSF responds to the audit data storage space exhaustion. This component traces back to and aids in meeting the following objectives: O.SELPRO, O.SECFUN, and O.SECSTA.

FMT_MOF.1Management of security functions behavior

This component ensures that the TSF restricts the ability to modify the behavior of functions such as audit trail management, backup and restore for TSF data, and communication of authorized external IT entities with the TOE to an authorized Administrative-user. This component traces back to and aids in meeting the following objectives: O.SECFUN, O.LIMEXT, and O.SECSTA.

FTA_SSL_EXP.3 TSF-Initiated Termination

This component ensures that the TSF terminates unattended GUI sessions after an Administrative-user configured time of inactivity to prevent unauthorized session access. This component traces back to and aids in meeting the following objective: O.SESSION_TERM.

FTA_SSL.4 User-Initiated Termination

This component ensures that the TSF provides the administrative-user with the ability to terminate sessions to prevent unauthorized session access. This component traces back to and aids in meeting the following objective: O.SESSION_TERM.

FRU_FLT.1 Degraded Fault Tolerance

This component ensures that the TSF has the ability to preserve security functionality and secure operation during specified failure events. This component traces back to and aids in meeting the following objective: O.SAFE_FAIL.

FRU_PRS.1 Limited Priority of Service

This component ensures that the TSF mediated access to TOE backend server client is provided on the basis of the subject's assigned priority. This component traces back to and aids in meeting the following objective: O.PRIORITY.

FRU_RSA_EXP.1(1) Maximum quotas – TCP Connections

This component ensures that the TSF provides mechanisms to enforce maximum quotas on the number of TCP connections allowed for subjects to use simultaneously. This component traces back to and aids in meeting the following objective: O.RESOURCE_X.

FRU_RSA_EXP.1(2) Maximum quotas – TCP SYN queue entries

This component ensures that the TSF provides mechanisms to enforce maximum quotas on the amount of TOE memory usage which may be used at one time to prevent a DoS based on BIG-IP resource exhaustion. This component traces back to and aids in meeting the following objective: O.RESOURCE_X.

FTP_ITC.1 Inter-TSF trusted channel

This component ensures that the TSF protects management traffic in transit from any workstation to the management port. This component traces back to and aids in meeting the following objective: O.SECFUN

FMT_SCR_EXP.1 Scripting of Flow Control Rules

This component ensures that the TSF provides a programmable script based mechanism to develop custom flow control polices, perform traffic analysis and implement specified actions based on programmatic constructs and conditional statements. This component traces back to and aids in meeting the following objective: O.SCRIPT_FUNC.

FDP_PXY_EXP.1 Reverse Proxy

This component ensures that the TSF provide mechanisms that proxy connections made through the appliance in such a way as to mask backend server resources and associated IP addresses while preserving the security properties of data flows. This component traces back to and aids in meeting the following objective: O.PROXY.

6.4 Rationale for IT Security Requirement Dependencies

This section includes a table of all the security functional requirements and their dependencies and a rationale for any dependencies that are not satisfied.

Functional Component	Dependency	Included/Rationale
FMT_SMR.1	FIA_UID.1	Yes, via FIA_UID.2
FMT_SAE.1	FMT_SMR.1, FPT_STM.1	Yes
FMT_SMF.1	None	None
FIA_ATD.1	None	None
FIA_UID.2	None	None
FIA_AFL.1	FIA_UAU.1	Yes, via FIA_UAU.2
FIA_UAU.2	FIA_UID.1	Yes, via FIA_UID.2
FIA_UAU.5	None	None
FIA_SOS_EXP.1	None	None
FCS_CKM.1 (1), (2), (3)	FCS_COP.1, FCS_CKM.4	No, FCS_CKM.4
FCS_COP.1(1),(2),(3),(4), (5),(6),(7),(8)	FCS_CKM.1, FCS_CKM.4	No, FCS_CKM.4
FDP_IFC.1 (1), (2)	FDP_IFF.1	Yes
FDP_IFF.1 (1), (2)	FDP_IFC.1, FMT_MSA.3	Yes
FMT_MSA.1 (1)	FDP_IFC.1, FMT_SMF.1, FMT_SMR.1	Yes
FMT_MSA.1 (2)	FDP_IFC.1, FMT_SMF.1, FMT_SMR.1	Yes
FMT_MSA.1 (3)	FDP_IFC.1, FMT_SMF.1, FMT_SMR.1	Yes
FMT_MSA.3 (1)	FMT_MSA.1, FMT_SMR.1	Yes
FMT_MSA.3 (2)	FMT_MSA.1, FMT_SMR.1	Yes
FMT_MSA.3 (3)	FMT_MSA.1, FMT_SMR.1	Yes
FMT_MTD.1 (1), (2), (3), (4), (5), (6), (7)	FMT_SMR.1, FMT_SMF.1	Yes
FDP_RIP.1	None	None
FPT_STM.1	None	None
FPT_FLS.1	None	None
FPT_FLS_EXP.1	None	None
FPT_ITC.1	None	None

FPT_ITI.1	None	None
FPT_ITT.1	None	None
FAU_ARP.1	FAU_SAA.1	Yes
FAU_GEN.1	FPT_STM.1	Yes
FAU_GEN.2	FAU_GEN.1, FIA_UID.1	Yes via FIA_UID.2
FAU_SAA.1	None	None
FAU_SAA_EXP.1	None	None
FAU_SAR.1	FAU_GEN.1	Yes
FAU_SAR.3	FAU_SAR.1	Yes
FAU_STG.1	FAU_GEN.1	Yes
FAU_STG.3	FAU_GEN.1	Yes
FMT_MOF.1	FMT_SMR.1, FMT_SMF.1	Yes
FTA_SSL_EXP.3	None	None
FTA_SSL.4	None	None
FRU_FLT.1	FPT_FLS.1	Yes, via FPT_FLS.1 and FPT_FLS_EXP.1
FRU_PRS.1	None	None
FRU_RSA_EXP.1(1)	None	None
FRU_RSA_EXP.1(2)	None	None
FTP_ITC.1	None	None
FMT_SCR_EXP.1	FMT_MSA.1, FDP_IFF.1	Yes
FDP_PXY_EXP.1	None	None

Table 11: SFR Dependencies

6.5 Rationale for SFR Dependencies not met

FCS_COP.1(1), (2), (3), (4), (5), (6), (7), (8) FCS_CKM.1 (1), (2), (3)	FCS_CKM.4	The dependency of FCS_CKM.4 for FCS_COP.1(1), (2), (3), (4), (5), (6), (7), (8) and FCS_CKM.1 (1), (2), (3) is not required as the use of cryptography by the TOE is limited to negotiating SSL sessions (on behalf of backend servers), encrypting cookies, creating X.509 certificates or securing Administrative-user sessions, internal data transfer, and creating sessions keys for these purposes. RSA and DSA keys stored on the TOE (used to generate session keys) are protected by A.PHYSEC & A.NOEVIL assumptions which specify that the TOE is deployed in a physically secure location and accessed only by trusted Administrative-users.

Table 12: Unsatisfied SFR Dependencies

6.6 TOE Security Assurance Requirements

The assurance security requirements for this Security Target are taken from Part 3 of the CC standard. These assurance requirements compose an Evaluation Assurance Level 2 augmented (EAL 2 + ALC_FLR.2) as defined by the CC. The assurance components are summarized in the following table.

Assurance Class	Assurance components
ADV: Development	ADV_ARC.1 Security architecture description
	ADV_FSP.2 Security-enforcing functional specification
	ADV_TDS.1 Basic design
AGD: Guidance documents	AGD_OPE.1 Operational user guidance
	AGD_PRE.1 Preparative procedures
ALC: Life-cycle support	ALC_CMC.2 Use of a CM system
	ALC_CMS.2 Parts of the TOE CM coverage
	ALC_DEL.1 Delivery procedures
	ALC_FLR.2 Flaw Reporting Procedures
ASE: Security Target evaluation	ASE_CCL.1 Conformance claims
	ASE_ECD.1 Extended components definition
	ASE_INT.1 ST introduction
	ASE_OBJ.2 Security objectives
	ASE_REQ.2 Derived security requirements
	ASE_SPD.1 Security problem definition
	ASE_TSS.1 TOE summary specification
ATE: Tests	ATE_COV.1 Evidence of coverage
	ATE_FUN.1 Functional testing
	ATE_IND.2 Independent testing - sample
AVA: Vulnerability assessment	AVA_VAN.2 Vulnerability analysis

Table 13: Assurance Requirements: EAL 2 + ALC_FLR.2

6.7 Rationale for TOE Assurance Requirements

The chosen assurance level is consistent with the postulated threat environment. Specifically, the threat of malicious attacks aimed at discovering exploitable vulnerabilities is considered low, and the product will have undergone a search for obvious flaws. This is supported by the inclusion of the AVA_VAN.2 requirement.

7 TOE Summary Specification

7.1 TOE Security Functions

The TOE's security functionality is characterized through the following Security Functions:

- Security Audit
- Identification and Authentication
- Security Management
- Secure Communications
- Secure Traffic
- Protection of the TSF
- User Data Protection: Information Flow Control

7.1.1 Security Audit

The TOE provides a comprehensive audit capability that generates event records and provides an audit trail of TOE security function and traffic management events. Through the GUI interface or tmsh, Administrative-users may view audit logs and filter displayed results based on information categories contained in the audit record.

Auditing and logging functions are managed entirely by the Syslog daemon within the BIG-IP TMOS. The audit function may be configured through the Syslog utility on the BIG-IP appliance to send BIG-IP appliance log information to a remote logging host.

Identification and Authentication failures are logged providing a resource to determine if unauthorized personnel may be attempting to access TSF functions.

Audited events are placed in 5 separate types of logs based on the type of event.

- System Events – TMOS level events
- Packet Filter Events – relating to implementation of packet filter rules
- Local Traffic Events – traffic management/routing related events
- Audit –GUI command logging, tmsh command logging, and system events
- Application Security (PSM) logs – events based on configured security profiles

Selecting and Reviewing Events (FAU_SAR.1, FAU_SAR.3)

As described above, all logs (with the exception of /var/log/httpd/ssl_access_log) can be accessed through the GUI, tmsh, or both and are formatted for easy viewing of pertinent information. Refer to the individual log sections below for details.

Audited events are, in general, placed in separate types of logs based on the type of event. There are cases where multiple types of events are places in the same log file, and where what would appear to be the same type of event is placed in a different log file. The latter is because of the

way the command or function is processed internally.

Table 8: Log searching and sorting defines the search and sort parameters available in tmsh and the GUI, and which apply to which log views. Tmsh may be used on all log files, and the log file or a subset piped to grep for searching. There is no sort capability in tmsh. The GUI parameters for sorting are specific to each tab (log view); searching is done by a text search in the search box.

Local Traffic Events – (FAU_GEN.1)

The "Local Traffic Events" logs produce audit events for Local Traffic Management events such as:

- IP packet discard events due to exceptional circumstances or invalid parameters (such as a bad checksum)
- MCP/TMM configuration events
- Pool and node status change events
- Network events (layer 1)
- iRules script related events related to run-time iRules script processing, when specified in the iRule
- General TMM events such as TMM startup, shutdown, and failover
- Tcpdump startup and shutdown

Local Traffic Event logs include Timestamp, Host Name, Description, Service, and Status Code information within each audit record. Local traffic logs cannot be disabled.

Local Traffic Events logs are accessed through the GUI or tmsh and are formatted for easy viewing of pertinent information.

Packet Filter Events records – (FAU_GEN.1)

The "Packet Filter Events" log is configured to identify packet discard events (from unidentified or questionable packets) that may result from thwarted attacks in accordance with configured packet filter rules. Packet Filter Event records include Timestamp, Host Name, Description, Service, and Status Code information in each audit record generated.. Packet Filter logs cannot be disabled.

Packet Filter Events logs are accessed through the GUI and are formatted for easy viewing of pertinent information.

System Log records – (FAU_GEN.1)

System Log records are generated for TMOS level events within the appliance and include Timestamp, Host Name, Description, and Service information within each audit record. System logs cannot be disabled.

System Logs are accessed through the GUI or tmsh and are formatted for easy viewing of pertinent information.

Audit (type) Log records – (FAU_GEN.1, FAU_GEN.2)

Audit (type) logging logs messages whenever a BIG-IP appliance object, such as a virtual server or a load balancing pool, is configured; that is, created, modified, or deleted. There are three

ways that objects can be configured:

- By user action
- By system action
- By loading configuration data, either by the the system loading the data at boot or restart, or by the user explicitly loading a stored configuration.

Audit (type) Log records are generated for these administrative-user related transactions and are grouped in the Security Management web interface (GUI) (or tmsh) by Timestamp, Username, Transaction Type, and Event. These audit records are associated by the identified user that caused the event.

The Administrator and Resource Administrator roles can disable audit (command-function-type) log auditing which creates an audit record prior to shutting down the particular type of auditing, however, the Common Criteria Evaluated configuration recommends that audit logging is enabled and set to the minimum level specified in the Common Criteria Administrator Guidance. This is to ensure that the correct events are logged, per the specified SFRs. . It is possible (though not recommended) to disable GUI and tmsh command logging, but not to disable system event logging.

It's important to note that audit (type) log records can be recorded in log files other than /var/log/audit. However, only the audit (command-function-type) log records logged in /var/log/audit can be disabled.

Audit (type) Logs are accessed through the GUI or tmsh (with the exception of /var/log/httpd/ssl_access_log) and are formatted for easy viewing of pertinent information.

Application Security (PSM) log records – (FAU_GEN.1, FAU_SAA_EXP.1))

The PSM module generates application security logs for security profile violations based on configured security profiles in force. When a configured rule is matched indicating a potential security violation, an audit record is made of the event by the Syslog daemon.

Security checks for HTTP include monitoring HTTP protocol compliance, Parameter Length checking to protect against buffer overflow attacks and validation of HTTP response codes and methods to assure they are not associated with possible application attack patterns. Sensitive user data in the HTTP message body is protected by monitoring data patterns and triggering a security violation if a pattern detected matches configured sensitive user data patterns. Also through security violation monitoring, the TOE can require that traffic include mandatory HTTP headers for validation prior to routing and assure that only specified file type and HTTP responses are allowed for traffic traversing the TOE.

Security checks for SMTP include protocol compliance checks that assure the received SMTP traffic meets RFC 2821 guidelines. Additional checks are provided that trigger when a sender matches an entry in the disallowed senders list; when the transaction contains a method in the disallowed methods list of the security profile; when the client transaction does not match the domain's SPF (Sender Policy Framework) record; when the client transaction is not valid based on the sender's IP address and type A DNS records; and when rate limiting/traffic shaping is applied against a specific sender address.

Security checks for FTP include protocol compliance checks to assure FTP traffic meets the guidelines of RFCs: 959, 1579 and 3659. Additional checks are provided that trigger if the FTP

uses the name "Anonymous", the FTP request uses the passive mode or the active mode, and FTP command length/arguments exceeds the maximum length allowed in the configured security profile. In addition, if an FTP command is used that is not listed in the Allowed Command list in the security profile, a violation is triggered. A configured maximum number of login attempts is allowed and, when exceeded, this also triggers a security violation.

By default, the Protocol Security Module retains up to 500 log entries per violation on the local disk.

If the Protocol Security Module BLOCK option is configured for a potential security violation, the triggering request is blocked by the TOE.

Application security logs cannot be disabled.

Potential Security violation analysis and Security Alarms (FAU_ARP.1, FAU_SAA.1, FAU_SAA_EXP.1)

The TOE has two sources of protential security violations and alarms.

The first includes all logging except for the messages provided by PSM. Table 7: Events logged by BIG-IP (with the exception of PSM logs) describes the potential security violations logged by the TOE. and available for the administrative-user to specify when configuring email alerts. After an event is logged, the log message is passed to the alerting daemon, where it is compared against messages configured to have email sent. If the message matches, an email alert is sent.

One example of this is the Denial of Service attempts that trigger security violations.TCP based traffic is evaluated against a configured threshold of 16384 (default) TCP requests. If this threshold is exceeded, a security violation is triggered and the Administrative-user is notified via email.

The PSM module provides statistics and violation data about traffic requests that trigger HTTP and SMTP security violations. Based on the configured Security Profile, an Alarm flag is set based on criteria for a violation. When an incoming request triggers a violation, the Protocol Security Module logs the request, which is then available from the Statistics screen of the Protocol Security Module section of the security management GUI.

The important difference between PSM and the rest of the logging is that PSM monitors traffic and issues alerts and generates log messages directly when the event is detected, thus supported FAU_SAA_EXP.1). The rest of the logging only issues the alerts once the logging is complete, and on the basis of the log message generated, not directly on the event (thus supporting FAU_SAA.1). Also, PSM doesn't handle management or configuration logging, only traffic events.

Audit Records – Timestamps & Protection (FPT_STM.1, FAU_STG.1, FAU_STG.3)

The BIG-IP appliance maintains an internal time source within the appliance that is suitable for use as a reliable time reference for audit records. All audit records include time/date information generated when the message is received indicating when the event occurred.

Only users holding the Administrator role have access to audit files. Audit records stored on the BIG-IP appliance cannot be modified by any user.

The TOE appliance allocates 7 GB for audit storage. A periodic cron job runs every two minutes to check the audit trail storage partition. When the storage reaches over 90% full, the TOE logs a

warning and clears space by deleting the oldest records, allowing new records to be logged. The TOE supports passing log file data to a Syslog server in the Operational Environment to mitigate the possibility of losing audit records.

7.1.2 Identification and Authentication

There are two types of users defined in the BIG-IP operating environment.

Administrative-users are those with authority to change the BIG-IP configuration and manage the TOE. There is always one local administrative-user (also known as the default Administrative-user) which has the Administrator role. It is managed through either the GUI or tmsh. Additional Administrative-users may be defined locally or remotely. If locally, then either the GUI or tmsh may be used to manage them. If remotely, the remote authentication server is used to manage them, with the caveat that there must be a local definition for the user that includes the user name and role. That local user definition is managed just as a complete administrative-user definition. The roles that can manage administrative-user properties are Administrator and User Manager.

Traffic users are users defined specifically for authenticating traffic flows. If all traffic flows are unauthenticated, no traffic users need be defined, If, however, any are defined, they are never defined locally. Instead, a remote authentication server must be used. Traffic user properties are stored by and managed through the remote authentication server only.

The BIG-IP Appliance has an internal authentication capability and can also be configured to use an external authentication server – LDAP or RADIUS, as described in section 1.8.2.

The default Administrative-user is required to be locally authenticated to assure access to the appliance is possible during a loss of connectivity to the external authentication server. When configured for external authentication, all other BIG-IP appliance users are authenticated using an external authentication server.

User Security Attributes for Identification and Authentication (FIA_ATD.1)

Table 14 User Attributes by User Type defines the specific attributes for Administrative users and traffic users.

Traffic user attributes, if any traffic uers are defined, are stored remotely only and include User name, User ID, User Password, and X.509 Distinguished Name. These attributes are managed remotely by a remote authentication server and not by the TOE.

Administrative-user attributes are stored within the TOE OS locally and within the TOE Environment remotely when applicable. User attributes include User name, User ID and User Password (in a MD5 hashed format). The Administrative partition identifier (as applicable) and User Role attributes are stored locally in the configuration database for all users.

Administrative-user	Traffic user

User name User ID User Password Administrative Partition User Role	User name User ID User Password X.509 Distinguished Name (DN)

Table 14 User Attributes by User Type

The BIG-IP TOE requires administrative users be positively identified and authenticated within the system prior to acquiring administrative access and/or performing any security functions. The TOE utilizes User accounts and roles to control access and manage privileges. In addition, Administrative-users may create Administrative Partitions to allocate certain configuration objects to specific storage locations within the BIG-IP system. Once configured, user must be assigned access to the applicable partition in addition to the access privileges granted through their role assignment in order to implement Administrative functions on partitioned objects..

The TOE constructs a secure channel via SSL with a client to be used for administrative access for access to the security management GUI interface, or via SSH for access to the tmsh interface. The secure channel is established only after each device authenticates itself.

Authentication by BIG-IP (FIA_UAU.2; FIA_UID.2; FIA_UAU.5)

For Local Authentication (administrative users), the TOE requires identification and authentication via a username and password combination. Authentication occurs internal to the TOE. Identification and Authentication is required prior to accessing TSF functions. By design, the default Administrative-user is always locally authenticated to ensure local access is always available to this full access Administrative role. The Administrator and User Manager can configure which Administrative-users are allowed to authenticate locally. Password hashes are securely stored in the OS using an MD5 hash including an 8 character salt. When a user needs to authenticate, they enter their password, the TOE hashes the password using the same salt, and if it matches, then the user is authenticated.

External authentication is accomplished by the use of LDAP or RADIUS servers.

Self Protection during Administrative-user sessions – FTA_SSL_EXP.3; FTA_SSL.4

The Administrative security management interface monitors GUI sessions for inactivity and terminates a session based on an Administrator or User Manager-configured time period of inactivity. If the TOE GUI application does not detect activity it begins a timer and after the configured amount of time has passed, it automatically logs the user out and terminates the session. By default, this value is set to 20 minutes. The GUI also provides a button on the user interface that allows users to terminate their sessions manually, if they plan to leave the area where the Administrator Console is located (tmsh users may manually terminate their ssh session). These features support protection of the TOE in that they assist in preventing unauthorized access by a passer-by in the event the Administrator Console workstation is unattended.

Remote Authentication by Authentication Server

For Remote Authentication, a remote authentication server is utilized. The types of remote authentication servers used for storing User accounts for BIG-IP are: Lightweight Directory Access Protocol (LDAP) servers and Remote Authentication Dial-in User Service (RADIUS.

For Administrative-users, the authentication happens as part of the login via the GUI or tmsh SSH sessions, or over HTTPS..

For traffic users, the traffic user to be authenticated is presented with a logon screen. Upon entering credentials on the provided screen, the User is identified and authenticated for access to backend resources requiring authentication. Administrative-users are responsible for ensuring that the password policies pertaining to the format of the password are enforced by remote authentication servers; the TOE cannot enforce that. The TOE, however, does ensure that authentication failure tracking and lockout occurs.

Security of TSF data transfer with External Authentication Server

(FPT_ITC.1; FPT_ITI.1, FCS_COP.1(5), FCS_COP.1(7))

Communication with the authentication server (trusted IT product) is secured through based on the preferred method of the external authentication server. RADIUS uses an MD5 password verification scheme, and LDAP uses SSL. to assure that TSF data such as username, passwords, or other authentication data is not disclosed to unauthorized parties, modified, or deleted. The TOE can detect a single Message Authentication Code (MAC) error during transmission and upon detecting this error will execute a resend command.

Use of the external authentication server can be limited by the Administrator or User Manager who can establish which User accounts must reside locally on the TOE and are not allowed to reside on the remote authentication server through configuration options.

The TOE receives the remote authentication request (username & password) and routes that request to the remote authentication server for validation. Prior to successful validation of credentials, no access to TOE TSF is allowed.

User roles are maintained in the local TOE database. Following identification and authentication via a remote server, the role information and partition identifier (if applicable) is accessed locally for the identified user.

Password Policy for Common Criteria Evaluated Configuration (FIA_SOS_EXP.1)

The minimum password policy enforced by BIG-IP through technical means for locally-authenticated Aministrative-users (except for the Administrator and User Manager roles) requires at least 8 characters, and at least one from each of the following: capital letters, lowercase letters, numbers, and punctuation. The following is the set of available characters for password selection:

- (alpha)A-Z a-z
- (numeric) 0-9
- (special characters) `~!@#$%^&*()-_+=[]{};’:”,./<>?|\

This set includes:

52 alphabetic characters (26 upper and 26 lower)

10 digits
10 punctuation marks from the shifted digits
22 more punctuation marks from other keys
For a total of 94 characters.

In addition, the password must meet the following criteria:

- The password must not be based on the userid or password entry, or be derived or derivable from the password entry
- The password must not be based on a dictionary word or reversed dictionary word, as defined by the systems dictionaries included with the Linux PAM module
- The password must not match a former user password kept in password memory (configured as holding 0-127 former passwords per user) unless password memory is cleared by having an authorized administrative-user set the password
- The password must be valid for a minimum duration, a maximum duration, and the user must be given a warning before the password expires. All values must be set as appropriate for the environment in which the TOE is installed. The configured values are:
    - Minimum duration = 0 days
    - Maximum duration (password expiration) = 90 days
    - Password expiration warning = 7 days.

This password policy applies to all locally-authenticated Administrative-users except for the roles: Administrator and User Manager. Guidance documentation instructs Administrator and User Manager role users to adhere to this policy on a procedural basis. Note that this applies to all local user passwords set by the Administrator and User Manager, whether their own or other users'.

Authentication Failure Handling (FIA_AFL.1)

An Administrator or User Manager may configure the number of failed logins before user accounts are disabled. This number can be from 0-65535. A value of 0 means this feature is disabled; however the evaluated configuration requires that the administrative user not disable this feature.

Once the account is disabled, no further login attempts are processed. The Administrative-user must re-enable the account in order to allow for additional login attempts to proceed. The login failures and account lock out are audited through the security audit security function. This feature is implemented by the security management application with the PAM module, and applies to both local and remote authentication. In the case of remote authentication, the TOE checks the return from the external authentication server for success or failure; subsequent processing is identical to that for local authentication with respect to tracking and acting on failures.

In addition to the failed login lockout described above, after each failed authentication, there is a delay of 2 seconds. This delay reduces the number of authentication attempts that can be made through automated means, making a brute force attempt more time intensive in the event the authentication failure mechanism was not engaged.

The default administrative user cannot be locked out.

PAM Module

The PAM (Pluggable Authentication Module) is part of the Linux operating system architecture which allows for customization of authentication checking. BIG-IP has a PAM that enforces the user policy rules described above. In addition, deployed external authentication servers also add a PAM to the system so that they can be accessed when external authentication is required.

Certificate based authentication for SSL sessions (FIA_UAU.5)

Leveraging the installed Advanced Client Authentication module, the BIG-IP appliance supports authentication of Administrative-user communications and of Client/Server connections using X.509 certificates.

Certificates for Administrative-user communications are created and self-signed within the BIG-IP appliance.

Certificates for Client/Server connections can be created and self signed within the BIG-IP appliance or imported when signed by a Certification Authority (CA). When BIG-IP receives requests from a Client or Backend Server, it can require that a CA-signed certificate be presented and then validate that certificate using an OCSP server in the Operational Environment. The following describes how the TOE validates certificates for SSL sessions.

Client-side certificate verification

When a client presents a certificate to the BIG-IP appliance (as it would to a backend server), the TOE uses a client trusted CAs file to determine if the presented certificate has been signed by a CA configured as trusted in the corresponding Client profile. BIG-IP then verifies the revocation status of the presented certificatean OCSP server in the Operational Environment.

Server-side certificate verification

When a Server SSL profile is enabled and the option to require a Server-side certificate is selected, the BIG-IP will perform Server-side certificate verification. When a server presents a certificate to the BIG-IP appliance, the TOE access the Server trusted CAs file to verify that the certificate is signed by a valid authority as configured in the Server-side profile. Once the CA verification step is complete, the TOE verifies the revocation status using the same method noted above using the OCSP server in the Operational Environment.

The parameters configured as part of an SSL profile can be found in the Configuration Guide for BIG-IP Local Traffic Manager, Chapter 10 (Managing SSL Traffic), section "Configuring SSL Profile Settings."

7.1.3 Security Management

The BIG-IP appliance provides a GUI based security management interface for configuration, maintenance and operational management purposes. All Administrative-users have access to the management interface based on the privileges allocated to them through their assigned role. Administrative-users may also manage the TOE from the TOE's commnd line utility, tmsh, via an SSH session. Most management functions can be performed through either the GUI or

tmsh,but some can only be configured from one or the other interface. Two examples of this are the PSM functions, which can only be configured by the GUI, and the syslog server, which can only be configured by tmsh.

The GUI is provided through an Apache HTTP web server implementation on the BIG-IP appliance and works directly with an installed Tomcat Java Servlet engine that renders web pages through the Apache web service to administrative-users running browser sessions on a management console machine in the Operational Environment. All sessions are secured using SSL sessions and AES encryption. The HTTPd service also interacts with the PAM authentication module within the underlying OS, for the purpose of authenticating users to the TOE. Further information regarding authentication is provided in Section 7.1.2.

Access to tmsh is provided through the SSH implementation on the BIG-IP appliance. Sessions using tmsh are performed using an ssh client from the administrator console machine in the Operational Environment connected to the management port on the BIG-IP. These SSH sessions are secured using the SSH protocol and AES encryption. SSH also interacts with the PAM authentication module within the underlying OS for the purpose of authenticating users to the TOE.

Security Management within the TOE is managed by the underlying TMOS through User Access controls and role based privileges. The TOE provides the Security Management security function to allow the administrative-user to configure security attributes which support the AUTHENTICATED SFP. Role based user access controls restrict the ability to query, modify, or delete the following security attributes to the Administrator and User Manager roles: User Definitions, Password Policy settings, Administrative partition identifier (when applicable) and Role Assignments.

The TOE supports a series of security management functions as specified in FMT_SMF.1. These include audit/logging management and review, general appliance configuration, flow control/security policy settings, user management and appliance backup utilities.

iRules Overview (FMT_SCR_EXP.1)

An iRule is a scripting language used to create scripts that allow the TOE Administrative-user to augment BIG-IP Information Flow Control by programmatically recognizing an event occurring in a traffic flow and performing an operation based on the available Information Flow Control objects. For example, the iRule could configure individual connections to target a pool other than the default pool defined for a virtual server, sending traffic not only to pools, but also to individual pool members, ports, or URIs given specified conditions.

Since iRules is a scripting security management tool, iRules scripts can be written to perform a wide variety of functions based on the script programmed. Examples of typical iRules script based implementations include:

- Selecting traffic destination based on scripted parameter
- Query header or content and return data
- Perform data manipulation including insertion of headers into HTTP requests
  Parse and manipulate content including decoding and return result
- Redirect, insert of transform application content (content transformation gateway)

- Perform a search on packet headers using scripted parameters, perform analysis based on scripted parameters and direct packets based on configured parameters
- Perform a search of packet contents using scripted parameters perform analysis based on scripted parameters and direct packets based on configured parameters
- Direct traffic to configured Pools, Individual pools members, specific ports, URI paths in order to implement persistence settings or load balancing objectives
- Detecting sensitive data patterns in the HTTP message body and inserting null data into the data stream to sanitize sensitive data flows (SSN from 123-45-6789 to xxx-xx-xxxx) aka Resource Cloaking

  iRules scripts are constructed by creating scripts which specify a given event, a conditional rule set and an action to take given the condition. The syntax used to write iRules scripts is based on the Tool Command Language (TCL) programming standard. Thus, the Administrative-user can use many of the standard TCL commands, plus a robust set of extensions that the TOE provides to create these scripts.

  iRules scripts are made up of: Event declarations, Operators, and Commands as described below:

  Event Declarations

  An event declaration is the specification of an event within an iRule script that causes the BIG-IP system to trigger that iRule script whenever that event occurs. A summary listing is provided below and a comprehensive listing is provided in Appendix B.

Global Events:

- CLIENT_ACCEPTED Triggered when a client establishes a connection.
- CLIENT_DATA Triggered when a client receives new data while the connection is in collect state.
- SERVER_SELECTED Triggered when the system has selected a target node.
- SERVER_CONNECTED Triggered when the system establishes a connection with the target node.
- SERVER_DATA Triggered when the system has received new data from the target node while the connection is in hold state.
- RULE_INIT Triggered when an iRules script is added or modified. This event is used to initialize global variables used within iRules scripts.

HTTP Events:

- HTTP_REQUEST Triggered when the system fully parses a complete client request header (that is, the method, URI, version and all headers, not including the body).
- HTTP_REQUEST_DATA Triggered whenever the request receives new
- HTTP content data.
- HTTP_RESPONSE Triggered when the system parses all of the response status and

header lines from the server response.

- HTTP_RESPONSE_DATA Triggered whenever the system receives new HTTP content data from the response.
- HTTP_RESPONSE_CONTINUE Triggered whenever the system receives a **100 Continue** response from the server.

SSL Events:

- CLIENTSSL_HANDSHAKE Triggered when a client-side SSL handshake is completed.
- CLIENTSSL_CLIENTCERT Triggered when the system adds an SSL client certificate to the client certificate chain. The system can retrieve the X.509 certificate and its X.509 issuer with the SSL::cert and SSL::cert issuer commands.
- SERVERSSL_HANDSHAKE Triggered when a server-side SSL handshake is completed.

Authentication Events:

- AUTH_FAILURE Triggered when an unsuccessful authorization operation is completed. A default handler for this event is associated with each of the authentication profiles, and causes the system to close the connection.
- AUTH_ERROR Triggered when an error occurs during authorization. A default handler for this event is associated with each of the authentication profiles, and causes the system to close the connection. The associated authentication session ID is invalidated and the user should immediately discard the session ID upon receipt of this event.
- AUTH_WANTCREDENTIAL Triggered when an authorization operation needs an additional credential. A default handler for this event is associated with each of the authentication profiles, and causes the system to close the connection unless it can obtain the desired credential.
- AUTH_SUCCESS Triggered when a successful authorization has completed all of the required authentication services.

Operators

An iRule operator compares two operands in an expression. In addition to using the TCL standard operators, the following additional operators are supported:

Operator	Syntax

Relational operators	`contains`
	`matches`
	`equals`
	`starts_with`
	`ends_with`
	`matches_regex`

Logical operators	`not`
	`and`
	`or`

For instance, if an HTTP_REQUEST event was received, it could trigger an iRule script which contains a rule set based on Operators (If contains “x”, then “y”, else “z” etc) which when matched results in the execution of an iRule command (example), HTTP::header remove <name>, which strips the named header for a request or response.

Commands

An iRule command within an iRule script 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 supported within iRules for scripting 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.

A full listing of iRules commands is provided in Appendix B.

iRules and TMOS Security Management

The iRules scripting function allows for the control of connections passing through the Local Traffic Manager. Through this configurable iRules scripting function, security policies are established and assigned to defined profiles to further manage functionality of the TSF. iRules scripting functionality is accessible to Administrator, Resource Administrator, and Manager roles only.

iRules scripts may be created and assigned to virtual servers only by administrative-users with specific roles. Also, if created in a specific partition, iRules may only be associated with virtual servers in that partition. However, the iRules scripts themselves have access to objects from all partitions.

Security Management Functions (FMT_SMF.1)

Administrative-user access is required for managing all functions, including the following security management functions as detailed in FMT_SMF.1:

- Security Audit Management
- Startup/Shutdown TOE operation

- Management of SFP rule attributes and Information Flow rules
- Administrative – User Account Management
- Virtual LAN/Server Management, Node/Pool configuration
- Traffic Protocol Profile configuration including iRules scripts
- PSM Security Profile configuration/management
- Backup/Restore configuration data

Role based User Access (FMT_SMR.1)

Administrative-users manage TSF Access and associated administrative functions within the TOE using role based privileges. New Administrative-users are assigned the appropriate permissions by role through selections made by the Administrator or User Manager during initial User Configuration.

The following Administrative-user roles are available to limit access to TOE security functions (FMT_SMR.1) based on the assigned privilege level. By default, a new user is assigned to the No Access role until explicitly assigned a specific role by the Administrator or User Manager:

Administrator

This role grants users complete access to all partitioned and non-partitioned objects on the system. In addition, accounts with the Administrator role can perform configuration synchronization on a redundant system. Administrators may change their own passwords.

Resource Administrator

This role grants users complete access to all partitioned and non-partitioned objects on the system, except user account objects. These users can perform configuration synchronization on a redundant system. Resource administrators may change their own passwords.

User Manager

User Managers have access to assigned partitions and can create, modify, delete, and view all user accounts except those that are assigned the Administrator role or the User Manager role with different partition access. Users with the User Manager role that have access only to a single partition can create, modify, delete, and view only those user accounts that are in that partition and that have access to that partition only. User accounts with the User Manager role can change their own passwords.

Manager

This role grants users permission to create, modify, and delete virtual servers, pools, pool members, nodes, custom profiles, custom monitors, and iRules scripts. These users can view all objects on the system and change their own passwords.

Application Editor

This role grants users permission to modify nodes, pools, pool members, and monitors. These users can view all objects on the system and change their own passwords.

Operator

This role grants users permission to enable or disable nodes and pool members. These users can view all objects and change their own passwords.

Guest

This role grants users permission to view all objects on the system and change their own passwords.

Web Application Security Administrator, Web Application Security Editor

These roles grant permissions for configuring Application Security Module objects, for viewing all other objects, and for changing their own password.

No Access

This role prevents users from accessing the system.

User accounts, roles, and associated permissions are protected from unauthorized access through the PAM module functionality working with the BIG-IP TMOS. This allows no access to TOE security functions until properly authenticated. In addition to the role based permissions, Administrative partitions can be created by administrative-users that place specific configuration and traffic management objects in configured partitions, available only to those users associated with those partitions via an Administrative partition identifier.

Audit related attributes can be set only by the Administrator or Resource Administrator role through the logging screens in the GUI. (FMT_MOF.1)

Administrative Partitions

Administrative Partitions are logical containers that administrative-users may create to further refine which administrative-users may access which BIG-IP objects. By allocating eligible objects to a partition, individuals can be granted access to a specific group of objects. The group can be defined so that the individual may manage a subset of the BIG-IP objects he otherwise would be permitted to manage based on his role-based privilege level. By default, BIG-IP has a single "Common" partition where all objects are maintained. Administrators then can establish new Administrative partitions and move objects to created partitions based on the management scheme. Once implemented, Users who have permission to access all partitions can actively select the "current" or active partition for their session. If only a single Administrative partition is assigned to a user, then it is selected by default during login. The following object types are managed through Administrative partitions:

- User accounts
- Virtual servers
- Pools
- Pool members
- Nodes
- Custom profiles (incl. objects associated authentication profiles)
- Custom monitors
- Traffic classes
- iRules

Profiles

Profiles are provided within the BIG-IP appliance to allow authorized Administrative-users to configure application specific network traffic in specific ways based on protocol type.

Management of Security Function behavior (FMT_MOF.1)

Through the security management security function, authorized Administrative-users can access configuration options that allow the tailoring of security function behavior based on each deployment scenario. Access to these settings is controlled via role based access controls as described above.

The Administrator and Resource Administrator roles may disable, enable, or modify the behavior of the Audit security function. For example, the Administrative-user could enable or disable auditing of command functions (for troubleshooting etc), however, all other logs run by default and cannot be disabled. When command-function auditing is disabled or enabled, an audit record is generated to indicate the deactivation or activation of auditing.

In addition, the TOE includes a utility that allows Administrator or Resource Administrator users to create a backup of TSF data and configuration files to a file server in the Operational Environment. Backups may only be transferred off the TOE through the secured web GUI.Communication with the TOE by authorized IT Entities is managed by network configuration of the TOE and can be managed by Administrator, Resource Administrator, or Manager roles.

User related Security Function behavior and related settings can only be enabled/disabled by the Administrator or User Manager role, however, User Manager Administrative-users cannot alter settings associated with the Administrator's account. All roles with the exception of the No Access role may change their own password.

The TSF restricts the ability to enable or disabled security functions associated with the operational status of Nodes/Pool Members to the Administrator, Resource Administrator, Manager or Application Editor roles.

Authentication profiles are configurations used to implement a PAM authentication module, and may be managed by Administrators, Resource Administrators, or Manager roles. The PAM controls how the TOE interacts with external authentication servers (LDAP or RADIUS).

Management of Security Attributes – (FMT_MSA.1 (1); FMT_MSA.1 (2); FMT_MSA.3 (1), FMT_MSA.3(2), FMT_MSA.3(3))

Security attributes associated with the Security Function Policies which mediate traffic flow through the TOE may only be created, modified, or deleted by Administrative-users holding the role of Administrator, Resource Administrator, or Manager. This also applies to creating, modifying, or deleting attributes within a flow control rule used to enforce the SFP. This includes settings that indicate IP address ranges, Protocols allowed, Services allowed and application security feature settings for Protocol compliance checking and Administrative-user configurable security features such as file type association, mandatory header requirements, allowed response codes, and user data patterns to restrict to protect sensitive data.

Default behavior associated with such settings may only be altered by Administrator or Resource Administrator roles. By default, the TOE enforces restrictive settings on traffic flows through the appliance; unless explicitly configured to allow flow, traffic cannot traverse the TOE appliance. The exception to this is the tmsh configuration for virtual servers; when creating a virtual server

using tmsh and allowing the default to be used for the destination parameter, the virtual server will be created with any IP address and any port allowed.

Management of TSF data – (FMT_MTD.1(1); FMT_MTD.1(2); FMT_MTD.1(3), FMT_MTD.1(4); FMT_MTD.1(5); FMT_MTD.1(6); FMT_MTD.1(7); FMT_SAE.1)

Access to modify TSF data is restricted based on the specific type of data and the Administrative-user authenticated role. The roles and restrictions are documented in Section 6.2.4.4.

User-attribute data requires a little more explanation, In general, the Administrator role and the User Manager role both may create, delete, and modify users and user attributes, and no other roles may affect those attributes, There are several execptions to this.

- Passwords
  - Administrative-users with the administrator role may change any user's passwords.
  - User Managers may change any user's password except for those users with the administrator role.
  - Any user may change their own password.
- User Managers may affect any other user attribute, and create and delete users, except for users with the Administrator role.

Certificate Management

The BIG-IP TOE allows administrative-users to manage certificates on the appliance for the purpose of authenticating traffic between clients and backend servers typically as a part of SSL Termination with Client/Server verification. Through the security management interface the following certificate management functions can be performed by Administrators and Resource Administrators:

- Display information about all existing key pairs and certificates.
- Create requests for new key pairs and certificates and submit those requests to certificate authorities.
- Renew certificate requests.
- Display key and certificate properties.
- Import and export PEM-formatted keys and certificates.

For CA-managed certificates, Administrators and Resource Administrators may request that a revocation check be performed by an OCSP server in the Operational Environment.

7.1.4 Secure Communications

Administrative-user access to the TOE (FPT_ITC.1; FPT_ITI.1; FCS_COP.1 (2),(3),(4),(6),(8); FCS_CKM.1(1),(2); FTP_ITC.1)

The TOE provides secure communication channels for administrative users to access the TOE through an SSL protected web-based UI. A dedicated Gigabit management port is provided for remote web-based UI access. Web traffic is over port 443 and uses SSL to encrypt all web GUI

traffic. In addition, an Administrative-user can access the TOE's commnd line utility, tmsh, or via an SSH session.

No TSF data is accessible remotely prior to establishment of a valid SSL or SSH session.

Administrative sessions with the TOE appliance through the GUI are performed using a browser from the administrator console machine in the Operational Environment directly connected to the dedicated Administrator Management port on the BIG-IP. These HTTPS sessions are secured using the SSL protocol and AES encryption. Sessions using tmsh are performed using an ssh client from the administrator console machine in the Operational Environment connected to the management port on the BIG-IP. These SSH sessions are secured using the SSH protocol and AES encryption.

All authentication failures are logged providing a resource to determine if unauthorized personnel may be attempting to access TSF functions.

Administrator GUI based access requires connectivity to the management port through the dedicated LAN. Establishment of the SSL session is authenticated through the Administrative-user's username and password, and the BIG-IP's device certificate.

For SSL GUI access, the TOE uses uniquely generated 1024 bit RSA keys with self-signed certificates. Session keys are uniquely-generated AES 256 or 128 keys.

In addition, Administrative-users can access the TOE's command line utility, tmsh, securely via ssh. The TOE uses uniquely-generated 1024-bit RSA or DSA keys for securing this communication, and uniquely-generated AES 256 or 128 keys as session keys.

Communication between the TOE and the LDAP server is secured using SSL. The TOE uses uniquely-generated keys per the SSL column in Appendix A.

All GUI SSL and SSH keys are generated using the RNG supplied with the Linux 2.6 kernel. The random number generator gathers environmental noise from device drivers and other sources into an entropy pool. The generator also keeps an estimate of the number of bit of the noise in the entropy pool. From this entropy pool random numbers are created.

Both SSL and SSH use RSA or DSA (according to the definitions in Appendix A – Cryptographic Key Support) keys to sign and verify the secure communication. During session negotiation, those digital signatures are used to verify the server and optionally the client (depending on how the communication is configured).

Data transfer between sections of the TOE (FPT_ITT.1, FCS_COP.1.1 (7))

ConfigSync is the process used to keep the configuration information of the redundant-pair BIG-IPs that make up the TOE in sync. BIG-IP encrypts the UCS file used to transfer that information with a 256-bit AES key, and HTTPS is used to transfer the configuration information to the peer BIG-IP.

Note: The cryptography used in this product has not been FIPS certified nor has it been analyzed or tested to conform to cryptographic standards during this evaluation. All cryptography has only been asserted as tested by the vendor.

7.1.5 Secure Traffic

(FDP_IFC.1 (2), FDP_IFF.1 (2); FCS_CKM.1(1),(2),(3); FCS_COP.1(1), (3),(4),(5))

The TOE enforces requirements on traffic communications through the AUTHENTICATED SFP. Cookies passed between the BIG-IP appliance and backend clients are encrypted using AES, 192 bit keys to secure cookie data and to provide additional assurance as to the Client identity. When so configured the BIG-IP system encrypts HTTP cookies before sending them to the client system, which treats them as any other cookie, storing and including them on subsequent requests to the server, but otherwise not operating on them.

The TOE can encrypt BIG-IP persistence cookies as well as cookies that are embedded in the response from the server. Cookie encryption keeps information private if the cookie contains sensitive information about the web application to avoid information leakage.

When cookie encryption is enabled, the BIG-IP TOE extracts the unencrypted cookie from the server response, encrypts it using the AES cipher and encodes it using the Base64 encoding scheme. The TOE then embeds the encrypted cookie into the HTTP response to the client. On subsequent requests when the client presents the encrypted cookie to the BIG-IP, the BIG-IP appliance removes the cookie, decodes it using the Base64 encoding scheme, and decrypts it. The BIG-IP system then re-embeds the decrypted cookie in the HTTP request to the server. If cookie decryption fails, the encrypted cookie remains in the HTTP request as-is.

Note that there are known issues with cookie encryption:

1. The period (.) character is not allowed in cookie names. Only alphanumeric characters and the special characters dash and underscore (- and _) are allowed.
2. Client cookies not able to be decrypted by the BIG-IP which have a length of 4*n (where n=[0,1,2,…]) cause the connection to be reset. Cookies of other lengths (decryptable or not) are passed through to the server.

Cookie encryption keys use a passphrase based derived key derivation function (PBKDF). All passphrases are stored in the configuration in the BIG-IP filesystem and are not visible to users once entered (similar to user passwords).

In similar fashion, specific data objects or aspects of a Client session may be encrypted using this mechanism as well. Conditions can be configured to direct encryption resources to be applied to objects based on a variety of factors and improve performance of communications through the focused use of encryption on the most sensitive facets.

The transfer of data between the BIG-IP appliance and backend Servers (trusted IT products) is protected from disclosure, modification, or deletion through the flow control mechanisms enforced by the AUTHENTICATED SFP. When required, the BIG-IP appliance may be configured to encrypt traffic flows to backend server pools to provide added security during transit. For that secure traffic, in addition to the requirements above, the AUTHENTICATED SFP requires that a successful SSL session has been established through verified key exchange and certificate validation.

The TOE provides SSL termination functions (when so configured) which allows the BIG-IP Appliance to establish sessions between traffic users attempting to connect using SSL with backend web servers. During these HTTPS sessions, the BIG-IP decrypts the traffic, performs HTTP protocol analysis and security checks, and then routes the traffic to the configured

backend server resources. The BIG-IP performs the negotiation utilizing ciphers included in the SSL protocol (RFC 2246 for TLSv1.0, and the SSL v3.0 Specification for SSLv3). Secure security attributes reflecting the use and type of session keys are required by the TOE for SSL session cryptography settings. Optionally, the BIG-IP may also validate the certificate on behalf of the backend server; this is known as SSL Termination with Client/Server Verification.

SSL profiles are configured to define the parameters by which SSL sessions are managed. SSL profiles can be either client or server profile types. Once the SSL profile has verified that a client or server can be trusted, the BIG-IP system can then control the connection's access to the destination content. Details regarding the contents of an SSL profile are contained in Section 7.1.2.

The BIG-IP system features for SSL connections are:

- Inserting header fields into HTTP requests
- Connection persistence
- Limiting the number of concurrent client TCP connections to the maximum number of connections specified by the customer's SSL license
- Client authorization with an LDAP database server

SSL traffic session keys are generated using a hardware-based pseudo-random number generator which produces symmetric keys as listed within Appendix A.

RSA keys are generated by the RSA key generation algorithm using a hardware-based pseudo-random number generator. RSA key lengths of 1024, 2048, and 4096 are supported by the TOE for SSL traffic.

The TOE uses three components to cryptographically secure traffic.

- The Cavium Nitrox™ Security Macro Processor resides on a mezzanine board in the BIG-IP unit and provides hardware-assistance for SSL handshake processing, key generation, asymmetric RSA cryptographic operations, symmetric ciphers (AES) and cryptographic hashing functions (MD5, SHA-1).
- The TMM MicroKernel manages the SSL handshake and SSL record processing.
- The OpenSSL library component provides support to the TMM MicroKernel for X.509 certificate verification and signing.

The TOE manages SSL connections based on administrative-user established profiles. The two types of SSL profiles within the TOE are Client and Server.

- Client profiles allow the TOE to manage SSL connections for connections coming into the TOE from the (WAN) client system.
- Server profiles allow the TOE to process encryption tasks for connections being sent from the TOE to a target server.

Where appropriate, secure connections may be administrative-user configured to be re-encrypted within the TOE prior to routing to the client to maintain a secure channel at all times. This configuration is not required in the CC evaluated configuration.

When SSL termination is selected within the TOE, certificate verification and revocation checks

are executed within the TOE.

SSL session persistence may be enabled based on administrative-user configurable Client or Server SSL persistence profiles but is not required in the CC Evaluated Configuration. Note that if session persistence is enabled, any situation (for instance, the virtual server being disabled) that would cause a new connection not to be established does not necessarily break an existing connection. Packets will still be passed on the existing connection until the connection is removed from the connection table. This can occur either via connection idle timeout or explicit administrative-user command.

Note: The cryptography used in this product has not been FIPS certified nor has it been analyzed or tested to conform to cryptographic standards during this evaluation. All cryptography has only been asserted as tested by the vendor.

7.1.6 Protection of the TOE

Protection of TOE Flow Control (FDP_IFC.1 (1), (2); FDP_IFF.1 (1), (2))

The TOE OS protects itself through the Local Traffic Manager based on configurable settings. The system can be configured to alert Administrators to a suspected attack or TOE penetration attempt.

Local Traffic Manager Security features include:

- Packets that are determined to not meet protocol RFC requirements or appear to contain questionable content are rejected and discarded to protect TOE resources (aka Protocol Sanitization). In addition to the standard protocol checks, if the Protocol Security Module BLOCK option is configured for a potential security violation, the triggering request is blocked by the TOE.
- Authentication failures are logged providing a resource to determine if unauthorized personnel may be attempting to access TSF functions.
- Fragmented packets may be reassembled to stop fragmentation attacks.

Denial of Service Protection (FRU_RSA_EXP.1 (1); FRU_RSA_EXP.1 (2); FDP_IFF.1 (1), (2))

Availability and Denial of Service (DoS) protection is implemented through the use of two mechanisms: TCP SYN flood attack mitigation, and Memory protection configuration (both defined in FMT_SMF.1.1).

TCP SYN flood attack mitigation, which limits the number of simultaneous TCP SYN queue entries, is implemented using SYN cookies. When a predetermined threshold is reached for new or untrusted connections, the SYNCheck™ feature is activated, initiating the use of SYN cookies for subsequent TCP connections. BIG-IP discards the SYN queue entry after sending a SYN-ACK to alleviate the SYN queue and if the client is legitimate and responds with an ACK, the "cookie" value that is sent to the client allows BIG-IP to reconstruct the SYN and therefore continue to establish the TCP session. Since the SYN queue does not fill once the threshold has been reached this effectively prevents SYN ACK denial of service. The value for SYNCheck activation is set by default at 16834.

Memory protection configuration supports Denial of Service (DoS) protection by limiting the number of simultaneous TCP conntections using the Reaper function (aka Dynamic Reaping).

When the system reaches the Low Water Mark (default of 85% of memory) the TOE will begin aggressively reaping any unused or FIN connections. Once the system meets the Reaper High-Water Mark(default 95%), the system will not establish new connections until the memory usage drops below the "reaper low-water mark" to protect against DoS through resource exhaustion.

Failure in Secure State (FPT_FLS.1; FPT_FLS_EXP.1; FRU_FLT.1)

The Evaluated Configuration of the TOE is in the High Availability Redundant Pair configuration allowing for maximum availability under various failure conditions.

There are two methods that the TOE uses to ensure that state is preserved and that the redundant appliance can continue operation if the primary appliance fails.

The first is to ensure that the configuration of the TOE is synchronized across both parts of the redundant pair. This is accomplished manually; when the TOE is first configured, the administrative-user issues configuration commands through either the GUI or tmsh to copy the configuration from one appliance in the redundant pair to the other. After that, the same configuration commands are used after every configuration update to resynchronize the configurations. This satisfies FPT_FLS_EXP.

The second ensures that operational information kept in sync, failure of the primary of the redundant pair detected, and fail-over to the stand-by appliance occurs, all automatically. This satisfies FPT_FLS and FRU_FLT.

In the event a failure is detected, all traffic is transferred to the redundant appliance and the failed appliance is identified to the Administrative-user via audit logs as being in a failed state. Failures are detected through appliance health monitoring performed by the TMOS. The health monitor receives input from health checks made on system objects for purpose of deriving statistics about traffic flow through the device and determining optimal routing paths. Since both appliances are connected to the network this only involves a switching action occurs between the two appliances. The appliances maintain a "heartbeat" signal between them to monitor status.

The TOE preserves a secure state during the operational failure of a single TOE hardware/software device. Fail-Over is executed with minimal loss of traffic & TSF when configured in the Evaluated Configuration. The actual loss of connections depends on the average lifetime of a given connection and how many concurrent connection are established. Compared to the number of connections (even SSL connections) the box can support, the number lost is generally minimal.

The following features are available in a redundant-pair configuration:

Failover – defines behavior relating to protection of availability via failover redundancy functionality

System fail-safe – When System FailSafe is configured, the BIG-IP appliance monitors various hardware components, as well as the heartbeat of the adjacent appliance, and takes action if the system detects a failure. A delay in communication of the heartbeat signal of <200mS will result in the non-responding appliance being set to "inactive".

7.1.7 User Data Protection: Information Flow Control

The BIG-IP appliance enforces the UNAUTHENTICATED/AUTHENTICATED SFPs to assure that traffic flowing through the appliance is properly terminated and re-routed within the device, based on configured traffic management rules and routing parameters. The BIG-IP appliance mediates any IP protocol including HTTP, SMTP, and FTP traffic and provides routing based on traffic type, protocol, VLAN configuration settings and backend server availability. For HTTP and SMTP, the TOE can provide authentication support therefore both the AUTHENTICATED and UNAUTHENTICATED SFPs apply. For FTP traffic, the BIG-IP does not provide authentication support, therefore only the UNAUTHENTICATED SFP applies. In the event of encrypted HTTP traffic (HTTPS), the TOE has the ability to decrypt and then perform HTTP protocol and security checks using the SSL Termination functionality described in Section 7.1.5, Secure Traffic.

Included among the methods used for defining Information Flow Control rules is the iRules™ feature, which allows Administrators to script custom rule sets based on specific traffic management related events and threat related to their specific deployment scenario. See Section 7.1.3 for information relating to the iRules scripting capability and traffic management usage.

Flow Control Policy Enforcement (FDP_IFC.1 (1), (2); FDP_IFF.1 (1), (2); FRU_FLT.1; FRU_PRS.1, FDP_PXY_EXP.1)

Information Flow Control policies are configured per the UNAUTHENTICATED and AUTHENTICATED SFPs in the TOE to assure that traffic flows only to and from authorized sources/destinations. The AUTHENTICATED SFP includes an authentication requirement; mandating that client must be successfully authenticated through an external authentication server in the operational environment prior to being routed to configured backend server resources. The SFPs are primarily implemented through the configuration of the Security Profiles enforcing the core packet inspection/filtering functionality, configuration of VLANs and associated virtual servers on the appliance, and global examptions for individual packet filtering rules. The Information Flow Control Policies are augmented by application of configured iRules scripts.

A virtual server receives a client request, and instead of sending the request directly to the destination IP address specified in the packet header, sends it to any of several content servers that make up a load balancing pool. .

IP addressing to the internal network is mapped securely from External Addresses using SNAT internal to the BIG-IP appliance, thereby protecting internal network topology and IP addressing details from disclosure to the external network (reverse proxy). UDP/IP and TCP/IP packet headers containing the external IP Address are rewritten to reflect internal addresses where backend servers reside. As part of this process, any required translation from IPv4 format addresses to IPv6 format addresses (or vice-versa) is done. Packet headers are checksummed using a 16 bit CRC. UDP/IP and TCP/IP packets from the internal network destined for the external network are rewritten to external addresses by BIG-IP. This eliminates internal address details from outbound flows. As traffic is processed, security properties contained within data flows are preserved as each individual connection is managed through BIG-IP in its own context termed a “flow”.Each flow is handled in a single-threaded process to prevent multi-process access to the data, and thus the chance of data corruption.

Flow priorities for FRU_PRS are assigned based on TCP QoS values.

Since the TOE is deployed in a redundant-pair configuration it supports Flow Control policies even if a single TOE appliance fails. The redundant appliance periodically receives updated configuration data from the primary appliance and therefore can immediately implement the flow control policy in effect once failover has occurred.

Residual information protection (FDP_RIP.1)

As described in Section 1.8.7, the TOE deconstructs packets received and evaluates various characteristics to determine if a malicious process flow or attack is identified. Following this process and as necessary for VLAN routing, the TOE reconstructs packets for routing to and from backend servers. Memory locations used to store previous packet data that are used for padding and constructing new packet flows are completely cleared (both the old packet data and the rest of the memory buffer allocated for packet data) and overwritten with new packet data when the new packet is created.

Traffic Management Flow Control Rules (FDP_IFF.1 (1), (2))

The TOE supports flow control through Administrator configurable HTTP, SMTP and FTP security profiles that perform a series of security checks prior to allowing information flows to/from backend servers in the Operational Environment. For traffic flows where authentication is required, the AUTHENTICATED SFP includes an authentication requirement that must be met in addition to the applicable security profile; the UNAUTHENTICATED SFP does not include this requirement. The BIG-IP appliance supports authentication for HTTP and SMTP. Authentication for FTP is not supported through TOE mechanisms; therefore the UNAUTHENTICATED SFP applies for FTP traffic.

For HTTPS traffic, decryption is performed by BIG-IP (as described in the Secure Traffic security function) and is followed by HTTP protocol analysis and security checks as listed below.

HTTP Security Profiles

Through the PSM module, the TOE performs packet level inspections of HTTP traffic to assure that flow control rules are met. The collection of specific settings can be grouped into a Security Profile definition. The PSM module performs the following security checks on HTTP traffic using the default HTTP security profile:

- Validates HTTP protocol compliance
- Detects evasion techniques – checks coding methods
- Parameter Length Checking: URIs, query strings, POST data, and requests
- Validates HTTP methods
- Validates “file types”
- Enforces “mandatory headers”
- Masks sensitive data in responses with the Data Guard™ feature
- Validates acceptable response codes

HTTP Protocol validation

Protocol validation checks are the first conducted by the PSM module following processing of the request through the TMM. For SSL traffic flows (HTTPS), traffic is decrypted first and then processed as HTTP for protocol compliance and security checking. The HTTP request is evaluated to determine if it matches RFC related compliance aspects for HTTP traffic. As with other security checks, if the validation fails any aspect of Protocol validation an alarm and log entry is generated and the TOE blocks all requests not compliant with the HTTP protocol standard.

Coding method validation – detection of evasion techniques

For every HTTP request received a preprocessor implemented by the PSM detects coding methods for application attacks that are designed to avoid detection. These coding methods are known as evasion techniques. Evasion techniques trigger the Evasion technique detected violation. Examples of the techniques that the Security Enforcer detects include:

- Path traversal, for example, ..\..\..\
- Multiple backslash characters in a URI, for example, \\servername
- Bad unescape
- Bare byte decoding in a URI

The evasion techniques detection function runs by default and analyzes every request for evasion techniques and cannot be disabled. Only the blocking policy is configurable by the authorized administrative user.

Parameter Length Checking

The HTTP security profile defines maximum lengths for specified HTTP request components. This is intended to prevent buffer overflow attacks associated with those components. This feature allows the configuration of maximum lengths for the following HTTP request components:

- URIs
- Query Strings
- POST data
- HTTP request (full request)

HTTP Method Checking

The Protocol Security Module accepts GET, POST, and HEAD methods by default, however, any incoming HTTP request that includes a method other than those specified as allowed trigger a security violation. The authorized administrative user may configure additional methods based on the deployed environment to be allowed as necessary.

File Type Checking

The File Type checking function allows authorized Administrative-users to configure either an allowed or disallowed file type listing that will either explicitly allow or deny specified file types based on configuration.

HTTP Mandatory Header Checking

The Mandatory Header Checking feature allows authorized administrative users to specify custom header that must occur in every request in order for the request to be processed. If the request does not include the header listed in the security profile, the system registers a violation takes the configured action which includes Alarm, Block, or both.

Response Code Checking

The HTTP security profile may include response code checking that allows authorized administrative users to configure an allowed response code list that avoids common vulnerabilities in server responses. This can prevent responses from including sensitive user data (based on configurable user data attributes) in the HTTP message body or an invalid response code. If sensitive user data such as credit card numbers is detected, the Data Guard™ feature either (based on its configuration) replaces the sensitive data with asterisks (known as response scrubbing) or triggers a security violation and prevents any responses containing configured sensitive information.

Response code checking also protects against invalid response codes. For the HTTP security profile, the allowed response codes determine which response codes are acceptable within a server response. If the HTTP response code is in the 4XX range or the 5XX range, then only responses with a response code that appears in this list are returned as-is to the client.

If a response contains a response code other than those specified in the allowed response code list, the system issues the Illegal HTTP status in response violation and blocks the response as configured by the authorized Administrative-user.

The following table describes the HTTP security violation events which may be triggered by the TOE[16]:

[16] Note that the "Illegal URL length" violation refers to the length of the request component URL, not the full request URL

HTTP violation	Violation trigger event
Evasion technique detected	The format of the request contains encoding or formatting that represents an attempt to bypass security checks.
Information leakage detected	A server response contains sensitive user data.
Mandatory HTTP header is missing	The request does not contain an HTTP header specified as mandatory by the security profile.
HTTP protocol compliance failed	The request does not comply with the HTTP protocol compliance checks.
Illegal URL length	The incoming request references a URL whose length exceeds the acceptable length as specified in the security profile.
Illegal request length	The incoming request length exceeds the acceptable length as specified in the security profile.
Illegal query string length	The incoming request contains a query string whose length exceeds the acceptable length as specified in the security profile.
Illegal POST data length	The incoming request contains POST data whose length exceeds the acceptable length as specified in the security profile.
Illegal file type	The incoming request references a file type that is either not configured in the allowed file types list, or configured in the disallowed file types list, of the security profile.
Illegal method	The incoming request references a HTTP request method that either is not configured in the allowed methods list, or is configured in the disallowed methods list, of the security profile.
Illegal HTTP status in response	The server response contains an HTTP status code that is not configured in the allowed response codes list of the security profile.

Table 15: HTTP Violation Trigger Events

SMTP Security Profiles

Security checks are performed on SMTP traffic as mentioned above for HTTP traffic through the PSM module. The following security checks are conducted as part of an SMTP security profile:

- SMTP protocol compliance as defined in RFC 2821

SMTP violation	Violation trigger event
Disallowed Senders	In a Domain Name Service (DNS) reverse lookup that uses the client domain name, the client IP address resolves to a domain name that is configured as a disallowed sender in the security profile. If you have configured IP addresses in the disallowed senders list, then the Security Enforcer just compares the client IP address to the list.
Sender DNS	The client SMTP transaction contains an IP address that does not have an official domain name entry in the Domain Name Service (DNS). In other words, IP addresses that fail to have a successful reverse DNS lookup performed are considered unsafe senders.
Disallowed Users	The client SMTP transaction contains a user domain name that is configured as a disallowed user in the security profile.
User DNS	The client SMTP transaction contains an IP address or domain name that has neither an associated MX record nor an A record in the Domain Name Service (DNS).
Allowed Receivers	The client SMTP transaction contains a receiver who is not listed in the Allowed Receivers list in the security profile.
Rate Limit per Sender Domain	The number of SMTP transactions for the sender's domain exceeds the per minute rate that is specified in the security profile.
Rate Limit per Receiver Domain	The number of SMTP transactions for the receiver's domain exceeds the per minute rate specified in the security profile.
DNS SPF Record	The client SMTP transaction does not match the domain's SPF (Sender Policy Framework) record.
Directory Attack	The number of SMTP transactions for non-existent recipients exceeds the transactions per minute rate that is specified in the security profile.
SMTP Methods	The SMTP transaction contains a method that is in the disallowed methods list of the security profile.
Greylisting	A client sends the same message to the same recipient multiple times within the hold period that is specified in the security profile. If, after being rejected the first time, a message comes in within the specified hold period, it is also rejected.

Table 16: SMTP Trigger Events

FTP Security Profiles

Security checks are performed on FTP traffic through functionality provided by the PSM module. The following security checks are conducted as part of an FTP security profile:

FTP violation	Violation trigger event
Anonymous FTP Requests	The FTP request uses the FTP user name, **Anonymous**.
Active Mode	The FTP request uses the FTP active mode.
Passive Mode	The FTP request uses the FTP passive mode.
FTP Commands	The FTP command is not in the allowed FTP command list in the security profile.
Command Length Restriction	The length of the issued FTP command, including any arguments, exceeds the length specified in the security profile.
Maximum Login Retries	The number of logon attempts, either from a specific user name, or a specific host IP address, exceeds the maximum number specified in the security profile.
FTP Protocol Compliance Failed	The traffic does not comply with the FTP protocol checks, which are based on the FTP RFC documents.

Table 17: FTP Trigger Events

In addition to the security checks above, configured routing rules that must be met for the routing of HTTP traffic include that Network IP addresses must be validated within the BIG-IP, and configured iRules script requirements (if any) must be met for the applicable Pool members.

Header Inspection of Packets:

This feature allows the TOE to alter the flow of packets based upon headers or other criteria. This can be used to select a different pool based upon URI or to authenticate traffic before sending the request to the applicable servers.

Rules based Pool Selection:

Based on Administrator configurable iRules scripts (if any), traffic is routed to local servers based on speed, availability, or content. Various monitors are utilized to continuously measure availability and throughput for use in rules based real time routing decisions. These include health and performance based monitors.

8 Appendix A – Cryptographic Key Support

	GUI SSL / HTTPS (Also used for ConfigSync)	**Traffic SSL**	**Cookie Encryption**	**SSH CLI**
Asymmetric				
Key type / size	RSA 1024	RSA 1024, RSA 2048, RSA 4096	--	RSA 1024, DSA 1024
Generated by	Kernel RNG	HW RNG	--	Kernel RNG
Signing	Cert self-signed	Cert self-signed Also may be signed by customer CA (we generate CSR and send to the customer, and he can send it to his CA)	--	--
Symmetric				
Key type / size	Session AES 256, AES 128	Session AES 256, AES 128	AES 192	AES 256, AES 128
Generated by	Kernel RNG	HW RNG	PBKDF (passphrase based key derivation function)	Kernel RNG
Hash	SHA-1	SHA-1	HMAC MD5	HMAC MD5, SHA-1, hmac-ripemd160, ripemd160

9 Appendix B: iRules references

Master List of iRules commands:

GLOBAL::

accumulate, active_members, active_nodes, after, b64decode, b64encode, class, clientside, client_addr, client_port, clone, cpu, crc32, decode_uri, discard, domain, drop, event, findclass, findstr, forward, getfield, htonl, htons, http_cookie, http_header, http_host, http_method, http_uri, http_version, if, imid, ip_protocol, ip_tos, ip_ttl, lasthop, link_qos, listen, local_addr, log, matchclass, members, nexthop, node, nodes, ntohl, ntohs, peer, persist, pool, priority, rateclass, redirect, reject, relate_client, relate_server, remote_addr, return serverside, server_addr, server_port, session, sharedvar, snat, snatpool, static, substr, switch, table, tcl_platform, timing, translate, use, virtual, vlan_id, when, whereis,

AES::

AES::decrypt, AES::encrypt, AES::key,

AUTH::

AUTH::abort, AUTH::authenticate, AUTH::authenticate_continue, AUTH::cert_credential, AUTH::cert_issuer_credential, AUTH::last_event_session_id, AUTH::password_credential, AUTH::response_data, AUTH::ssl_cc_ldap_status, AUTH::ssl_cc_ldap_username, AUTH::start, AUTH::status, AUTH::subscribe, AUTH::unsubscribe, AUTH::username_credential, AUTH::wantcredential_prompt, AUTH::wantcredential_prompt_style, AUTH::wantcredential_type,

CACHE::

CACHE::accept_encoding, CACHE::age, CACHE::disable, CACHE::disabled, CACHE::enable, CACHE::expire, CACHE::fresh, CACHE::header, CACHE::headers, CACHE::hits, CACHE::payload, CACHE::priority, CACHE::statskey, CACHE::trace, CACHE::uri, CACHE::useragent, CACHE::userkey,

COMPRESS::

COMPRESS::buffer_size, COMPRESS::disable, COMPRESS::enable, COMPRESS::gzip, COMPRESS::method,

HTTP::

HTTP::class, HTTP::close, HTTP::collect, HTTP::cookie, HTTP::disable, HTTP::enable, HTTP::fallback, HTTP::header, HTTP::host, HTTP::is_keepalive, HTTP::is_redirect, HTTP::method, HTTP::password, HTTP::path, HTTP::payload, HTTP::query, HTTP::redirect, HTTP::release, HTTP::request, HTTP::request_num, HTTP::respond, HTTP::retry, HTTP::status, HTTP::uri, HTTP::username, HTTP::version,

IP::

IP::addr, IP::client_addr, IP::hops, IP::idle_timeout, IP::local_addr, IP::protocol, IP::remote_addr, IP::server_addr, IP::stats, IP::tos, IP::ttl, IP::version,

LB::

LB::class, LB::command, LB::detach, LB::down, LB::mode, LB::persist, LB::reselect, LB::select, LB::server, LB::snat, LB::status, LB::up,

LINK::

LINK::lasthop, LINK::nexthop, LINK::qos, LINK::vlan_id,

NAME::

NAME::lookup, NAME::response,

ONECONNECT::

ONECONNECT::detach, ONECONNECT::label, ONECONNECT::reuse, ONECONNECT::select,

PROFILE::

PROFILE::auth, PROFILE::clientssl, PROFILE::exists, PROFILE::fasthttp, PROFILE::fastL4, PROFILE::ftp, PROFILE::http, PROFILE::httpclass,

PROFILE::oneconnect, PROFILE::persist, PROFILE::serverssl, PROFILE::stream, PROFILE::tcp, PROFILE::udp, PROFILE::xml,

ROUTE::

ROUTE::age, ROUTE::bandwidth, ROUTE::domain, ROUTE::rtt, ROUTE::rttvar,

RTSP::

RTSP::collect, RTSP::header, RTSP::method, RTSP::msg_source, RTSP::payload, RTSP::release, RTSP::respond, RTSP::status, RTSP::uri, RTSP::version,

SCTP::

SCTP::client_port, SCTP::collect, SCTP::local_port, SCTP::mss, SCTP::payload, SCTP::ppi, SCTP::release, SCTP::remote_port, SCTP::respond, SCTP::server_port,

SDP::

SDP::field, SDP::media, SDP::session_id,

SIP::

SIP::call_id, SIP::discard, SIP::from, SIP::header, SIP::method, SIP::respond, SIP::response, SIP::to, SIP::uri, SIP::via,

SSL::

SSL::authenticate, SSL::cert, SSL::cipher, SSL::collect, SSL::disable, SSL::enable, SSL::handshake, SSL::mode, SSL::modssl_sessionid_headers, SSL::payload, SSL::profile, SSL::release, SSL::renegotiate, SSL::respond, SSL::session, SSL::sessionid, SSL::unclean_shutdown, SSL::verify_result,

STATS::

STATS::get, STATS::incr, STATS::set, STATS::setmax, STATS::setmin,

STREAM::

STREAM::disable, STREAM::enable, STREAM::encoding, STREAM::expression, STREAM::match, STREAM::max_matchsize, STREAM::replace,

TCP::

TCP::bandwidth, TCP::client_port, TCP::close, TCP::collect, TCP::local_port, TCP::mss, TCP::nagle, TCP::notify, TCP::offset, TCP::payload, TCP::release, TCP::remote_port, TCP::respond, TCP::rtt, TCP::server_port, TCP::unused_port,

UDP::

UDP::client_port, UDP::drop, UDP::local_port, UDP::mss, UDP::payload, UDP::remote_port, UDP::respond, UDP::server_port, UDP::unused_port,

URI::

URI::basename, URI::compare, URI::decode, URI::encode, URI::host, URI::path, URI::port, URI::protocol, URI::query,

X509::

X509::cert_fields, X509::extensions, X509::hash, X509::issuer, X509::not_valid_after, X509::not_valid_before, X509::serial_number, X509::signature_algorithm, X509::subject, X509::subject_public_key, X509::subject_public_key_RSA_bits, X509::subject_public_key_type, X509::verify_cert_error_string, X509::version, X509::whole,

XML::

XML::address, XML::collect, XML::element, XML::event, XML::eventid, XML::parse, XML::release, XML::soap, XML::subscribe

Informational Functions

These functions are the ones that will return a string of information (or a subset of a string), or a matching result.

- domain - Parses the specified string as a dotted domain name and returns the last **<count>** portions of the domain name.
- findclass - Searches a data group list for a member that starts with a specified string and returns the data-group member string.
- findstr - Finds a string within another string and returns the string starting at the offset specified from the match.
- getfield - Splits a string on a character or string, and returns the string corresponding to the specific field.
- matchclass - Performs comparison against a class.
- substr - Returns a sub-string named **<string>**.

Utility Functions

Utility functions are those commands that transform information and return the result in the desired format.

- b64decode - Returns a string that is base-64 decoded.
- b64encode - Returns a string that is base-64 encoded, or if an error occurs, an empty string.
- crc32 - Returns the crc32 checksum for the specified string.
- decode_uri - Decodes the specified string using HTTP URI encoding.
- sha1 - Returns the SHA version 1.0 message digest of the specified string.

Master List of iRule Events

AUTH	AUTH_ERROR, AUTH_FAILURE, AUTH_RESULT, AUTH_SUCCESS, AUTH_WANTCREDENTIAL,
CACHE	CACHE_REQUEST, CACHE_RESPONSE, CACHE_UPDATE,
CLIENTSSL	CLIENTSSL_CLIENTCERT, CLIENTSSL_DATA, CLIENTSSL_HANDSHAKE,
DNS	DNS_REQUEST, DNS_RESPONSE,
GLOBAL	LB_FAILED, LB_SELECTED, NAME_RESOLVED, PERSIST_DOWN, RULE_INIT,
HTTP	HTTP_CLASS_FAILED, HTTP_CLASS_SELECTED, HTTP_REQUEST, HTTP_REQUEST_DATA, HTTP_REQUEST_SEND, HTTP_RESPONSE, HTTP_RESPONSE_CONTINUE, HTTP_RESPONSE_DATA,
IP	CLIENTSSL_DATA, CLIENT_ACCEPTED, CLIENT_CLOSED,

	CLIENT_DATA, SERVERSSL_DATA, SERVER_CLOSED, SERVER_CONNECTED, SERVER_DATA,
LINE	
RTSP	RTSP_REQUEST, RTSP_REQUEST_DATA, RTSP_RESPONSE, RTSP_RESPONSE_DATA,
SCTP	CLIENTSSL_DATA, CLIENT_ACCEPTED, CLIENT_CLOSED, CLIENT_DATA, SERVER_CLOSED, SERVER_CONNECTED, SERVER_DATA,
SIP	SIP_REQUEST, SIP_REQUEST_SEND, SIP_RESPONSE,
SERVERSSL	SERVERSSL_DATA, SERVERSSL_HANDSHAKE,
STREAM	STREAM_MATCHED,
TCP	CLIENTSSL_DATA, CLIENT_ACCEPTED, CLIENT_CLOSED, CLIENT_DATA, SERVERSSL_DATA, SERVER_CLOSED, SERVER_CONNECTED, SERVER_DATA, USER_REQUEST, USER_RESPONSE,
UDP	CLIENT_ACCEPTED, CLIENT_CLOSED, CLIENT_DATA, SERVER_CLOSED, SERVER_CONNECTED, SERVER_DATA,
XML	XML_BEGIN_DOCUMENT, XML_BEGIN_ELEMENT, XML_CDATA, XML_END_ELEMENT, XML_EVENT,

AUTH EVENTS

- AUTH_ERROR - Triggered when an error occurs during authorization.
- AUTH_FAILURE - Triggered when an unsuccessful authorization operation is completed.
- AUTH_RESULT - Replaces AUTH_SUCCESS**,** AUTH_FAILURE**,** AUTH_ERROR**, and** AUTH_WANTCREDENTIAL **events.**
- AUTH_SUCCESS - Triggered when a successful authorization has completed all required authentication services.
- AUTH_WANTCREDENTIAL - Triggered when an authorization operation needs an additional credential.

CACHE EVENTS

- CACHE_REQUEST - Triggered when the system receives a request for a cached object.
- CACHE_RESPONSE - Triggered immediately prior to sending a cache response.
- CACHE_UPDATE - In Progress - Add Summary Here.

CLIENT SSL EVENTS

- CLIENTSSL_CLIENTCERT - Triggered when the system adds an SSL client certificate to the client certificate chain.
- CLIENTSSL_DATA - Triggered each time new SSL data is received from the client while the connection is in "collect" state.
- CLIENTSSL_HANDSHAKE - Triggered when a client-side SSL handshake is completed.

DNS EVENTS

- DNS_REQUEST - Triggered when the system receives a DNS request.
- DNS_RESPONSE - Triggered when the system responds to a DNS request.

GLOBAL EVENTS

- LB_FAILED - Triggered when the system fails to select a pool or a pool member, or when a selected resource is unreachable.
- LB_SELECTED - Triggered when the system selects a pool member.
- NAME_RESOLVED - Triggered after a NAME::lookup command has been issued and a response has been received.
- PERSIST_DOWN - Triggered when persistence dictates that a connection would be sent to a pool or a pool member or node which has been marked down.
- RULE_INIT - Triggered when an iRule is added or is modified.

HTTPS EVENTS

- HTTP_CLASS_FAILED - Triggered when an HTTP request is made to a virtual server with at least one HTTP class configured, and the request does not match the filters of any HTTP class.
- HTTP_CLASS_SELECTED - Triggered when an HTTP request matches an HTTP class.
- HTTP_REQUEST - Triggered when the system fully parses a complete client request header.
- HTTP_REQUEST_DATA - Triggered when an **HTTP::collect** command has collected the specified amount of request data.
- HTTP_REQUEST_SEND - Triggered immediately before an HTTP request is sent to the server-side TCP stack.
- HTTP_RESPONSE - Triggered when the system parses all of the response status and header lines from the server response.
- HTTP_RESPONSE_CONTINUE - Triggered whenever the system receives a **100 Continue** response from the server.
- HTTP_RESPONSE_DATA - Triggered when an **HTTP::collect** command has collected the specified amount of response data.

IP EVENTS

- CLIENTSSL_DATA - Triggered each time new SSL data is received from the client while the connection is in "collect" state.
- CLIENT_ACCEPTED - Triggered when a client has established a connection.
- CLIENT_CLOSED - This event is fired at the end of any client connection, regardless of protocol.
- CLIENT_DATA - Triggered each time new data is received from the client while the connection is in "collect" state.
- SERVERSSL_DATA - Triggered when new SSL data is received from the target node after SSL::collect command has been issued.
- SERVER_CLOSED - This Event is fired when the Server side connection closes.
- SERVER_CONNECTED - Triggered when a connection has been established with the target node.
- SERVER_DATA - Triggered when new data is received from the target node after TCP::collect command has been issued.

RTSP EVENTS

- RTSP_REQUEST - Triggered after a complete request has been received from either the client or the server.
- RTSP_REQUEST_DATA - Triggered whenever an RTSP::collect command finishes processing.
- RTSP_RESPONSE - Triggered after a complete response has been received from either the client or the server.
- RTSP_RESPONSE_DATA - Triggered when collection of response data is finished.

SCTP EVENTS

- CLIENTSSL_DATA - Triggered each time new SSL data is received from the client while the connection is in "collect" state.
- CLIENT_ACCEPTED - Triggered when a client has established a connection.
- CLIENT_CLOSED - This event is fired at the end of any client connection, regardless of protocol.
- CLIENT_DATA - Triggered each time new data is received from the client while the connection is in "collect" state.
- SERVER_CLOSED - This Event is fired when the Server side connection closes.
- SERVER_CONNECTED - Triggered when a connection has been established with the target node.
- SERVER_DATA - Triggered when new data is received from the target node after TCP::collect command has been issued.

SIP EVENTS

- SIP_REQUEST - Triggered when the system fully parses a complete client SIP request header.
- SIP_REQUEST_SEND - Triggered immediately before a SIP request is sent to the server-side TCP stack.
- SIP_RESPONSE - Triggered when a SIP Response is received from the Server.

SERVER SSL EVENTS

- SERVERSSL_DATA - Triggered when new SSL data is received from the target node after SSL::collect command has been issued.
- SERVERSSL_HANDSHAKE - Triggered when a server-side SSL handshake is completed.

STREAM EVENTS

- STREAM_MATCHED - Triggered when a stream expression matches.

TCP EVENTS

- CLIENTSSL_DATA - Triggered each time new SSL data is received from the client while the connection is in "collect" state.
- CLIENT_ACCEPTED - Triggered when a client has established a connection.
- CLIENT_CLOSED - This event is fired at the end of any client connection, regardless of protocol.
- CLIENT_DATA - Triggered each time new data is received from the client while the connection is in "collect" state.
- SERVERSSL_DATA - Triggered when new SSL data is received from the target node after SSL::collect command has been issued.
- SERVER_CLOSED - This Event is fired when the Server side connection closes.
- SERVER_CONNECTED - Triggered when a connection has been established with the target node.
- SERVER_DATA - Triggered when new data is received from the target node after TCP::collect command has been issued.
- USER_REQUEST - Triggered by command TCP::notify request.
- USER_RESPONSE - Triggered by command TCP::notify response.

UDP EVENTS

- CLIENT_ACCEPTED - Triggered when a client has established a connection.
- CLIENT_CLOSED - This event is fired at the end of any client connection, regardless of protocol.
- CLIENT_DATA - Triggered each time new data is received from the client while the connection is in "collect" state.
- SERVER_CLOSED - This Event is fired when the Server side connection closes.

- SERVER_CONNECTED - Triggered when a connection has been established with the target node.
- SERVER_DATA - Triggered when new data is received from the target node after TCP::collect command has been issued.

XML EVENTS

- XML_BEGIN_DOCUMENT - Triggered before the XML document gets parsed.
- XML_BEGIN_ELEMENT - Triggered when the parser has encountered the start of an element.
- XML_CDATA - Triggered when the parser has encountered character data (CDATA).
- XML_END_ELEMENT - Triggered when the parser has encountered the end of an element.
- XML_EVENT - A generic "catch-all" event that is triggered for all XML events.

ALL EVENTS

- ASM_REQUEST_BLOCKING - Triggered when ASM is generating the reject-response and gives the iRule a chance to modify that reject-response before it is sent.
- ASM_REQUEST_VIOLATION - Triggered when ASM detects that a request violates an ASM security policy.
- ASM_RESPONSE_VIOLATION - Triggered when ASM detects that a response violates an ASM security policy.
- AUTH_ERROR - Triggered when an error occurs during authorization.
- AUTH_FAILURE - Triggered when an unsuccessful authorization operation is completed.
- AUTH_RESULT - Replaces AUTH_SUCCESS**,** AUTH_FAILURE**,** AUTH_ERROR**, and** AUTH_WANTCREDENTIAL events.
- AUTH_SUCCESS - Triggered when a successful authorization has completed all required authentication services.
- AUTH_WANTCREDENTIAL - Triggered when an authorization operation needs an additional credential.
- CACHE_REQUEST - Triggered when the system receives a request for a cached object.
- CACHE_RESPONSE - Triggered immediately prior to sending a cache response.
- CACHE_UPDATE - In Progress - Add Summary Here.
- CLIENTSSL_CLIENTCERT - Triggered when the system adds an SSL client certificate to the client certificate chain.
- CLIENTSSL_DATA - Triggered each time new SSL data is received from the client while the connection is in “collect” state.
- CLIENTSSL_HANDSHAKE - Triggered when a client-side SSL handshake is completed.

- CLIENT_ACCEPTED - Triggered when a client has established a connection.
- CLIENT_CLOSED - This event is fired at the end of any client connection, regardless of protocol.
- CLIENT_DATA - Triggered each time new data is received from the client while the connection is in “collect” state.
- DNS_REQUEST - Triggered when the system receives a DNS request.
- DNS_RESPONSE - Triggered when the system responds to a DNS request.
- HTTP_CLASS_FAILED - Triggered when an HTTP request is made to a virtual server with at least one HTTP class configured, and the request does not match the filters of any HTTP class.
- HTTP_CLASS_SELECTED - Triggered when an HTTP request matches an HTTP class.
- HTTP_REQUEST - Triggered when the system fully parses a complete client request header.
- HTTP_REQUEST_DATA - Triggered when an **HTTP::collect** command has collected the specified amount of request data.
- HTTP_REQUEST_SEND - Triggered immediately before an HTTP request is sent to the server-side TCP stack.
- HTTP_RESPONSE - Triggered when the system parses all of the response status and header lines from the server response.
- HTTP_RESPONSE_CONTINUE - Triggered whenever the system receives a **100 Continue** response from the server.
- HTTP_RESPONSE_DATA - Triggered when an **HTTP::collect** command has collected the specified amount of response data.
- LB_FAILED - Triggered when the system fails to select a pool or a pool member, or when a selected resource is unreachable.
- LB_SELECTED - Triggered when the system selects a pool member.
- NAME_RESOLVED - Triggered after a NAME::lookup command has been issued and a response has been received.
- PERSIST_DOWN - Triggered when persistence dictates that a connection would be sent to a pool or a pool member or node which has been marked down.
- RTSP_REQUEST - Triggered after a complete request has been received from either the client or the server.
- RTSP_REQUEST_DATA - Triggered whenever an RTSP::collect command finishes processing.
- RTSP_RESPONSE - Triggered after a complete response has been received from either the client or the server.
- RTSP_RESPONSE_DATA - Triggered when collection of response data is finished.

- RULE_INIT - Triggered when an iRule is added or is modified.
- SERVERSSL_DATA - Triggered when new SSL data is received from the target node after SSL::collect command has been issued.
- SERVERSSL_HANDSHAKE - Triggered when a server-side SSL handshake is completed.
- SERVER_CLOSED - This Event is fired when the Server side connection closes.
- SERVER_CONNECTED - Triggered when a connection has been established with the target node.
- SERVER_DATA - Triggered when new data is received from the target node after TCP::collect command has been issued.
- SIP_CLIENT_MSG - In Progress - Add Summary Here.
- SIP_REQUEST - Triggered when the system fully parses a complete client SIP request header.
- SIP_REQUEST_SEND - Triggered immediately before a SIP request is sent to the server-side TCP stack.
- SIP_RESPONSE - Triggered when a SIP Response is received from the Server.
- SIP_SERVER_MSG - In Progress - Add Summary Here.
- STREAM_MATCHED - Triggered when a stream expression matches.
- USER_REQUEST - Triggered by command TCP::notify request.
- USER_RESPONSE - Triggered by command TCP::notify response.
- XML_BEGIN_DOCUMENT - Triggered before the XML document gets parsed.
- XML_BEGIN_ELEMENT - Triggered when the parser has encountered the start of an element.
- XML_CDATA - Triggered when the parser has encountered character data (CDATA).
- XML_END_ELEMENT - Triggered when the parser has encountered the end of an element.
- XML_EVENT - A generic "catch-all" event that is triggered for all XML events.

