POSITIVE TECHNOLOGIES

PT Application Firewall

Common Criteria Certification

Security Target

Author:	Egor Sorokin, Positive Technologies
Category:	CC Certification
Version:	2.9
Date:	2020-02-18
File Name:	PT_ST_2.9.pdf

Abstract

This document is the ST (Security Target) of the PT Application Firewall Common Criteria Certification.

Keywords

CC, ST, Common Criteria, Security Target

Prepared by

POSITIVE TECHNOLOGIES

Positive Technologies
Room 30, office V, Schelkovskoe shosse 23A
Moscow 107241
Russian Federation

Phone: +7 (495) 744-0144

https://www.ptsecurity.com

Table of Contents

Page

1 ST INTRODUCTION ... 6
1.1 Security Target and TOE references ... 6
1.2 TOE overview ... 7
1.2.1 Product Type ... 9
1.2.2 TOE Major Security Features ... 9
1.2.3 Description of the TOE components... **Ошибка! Закладка не определена.**
1.2.4 NON-TOE Hardware/Software/Firmware ... 9
1.3 TOE description ... 11
1.3.1 Physical scope ... 11
1.3.2 Logical scope and boundary ... 12
1.3.3 TOE Evaluated Configuration ... 12
1.4 Conventions ... 13
2 CONFORMANCE CLAIMS ... 15
2.1 CC conformance claims ... 15
2.2 PP claim ... 15
2.3 Package claim ... 15
2.4 Conformance rationale ... 15
3 SECURITY PROBLEM DEFINITION ... 16
3.1 Assets ... 16
3.2 Subjects ... 16
3.3 Threats ... 16
3.4 Organizational security policies ... 17
3.5 Assumptions ... 17
4 SECURITY OBJECTIVES ... 19
4.1 Security objectives for the TOE ... 19
4.2 Security objectives for the operational environment ... 20
4.3 Security objectives rationale ... 20
5 EXTENDED COMPONENTS DEFINITION ... 23
5.1 Extended TOE Security Functional Components ... 23
5.1.1 Class EXT_WAF: Web Application Firewall ... 23
5.2 Extended TOE Security Assurance Components ... 26
6 SECURITY REQUIREMENTS ... 27
6.1 Security functional requirements ... 27
6.1.1 Class FAU – Security Audit ... 28
FAU_GEN.1 AUDIT DATA GENERATION ... 28
FAU_GEN.2 USER IDENTITY ASSOCIATION ... 30
FAU_SAR.1 AUDIT REVIEW ... 30
FAU_SAR.2 RESTRICTED AUDIT REVIEW ... 30

6.1.2 Class FIA – Identification and authentication 30
FIA_ATD.1 USER ATTRIBUTE DEFINITION 30
DEPENDENCIES: NO DEPENDENCIES. 30
FIA_UAU.2 USER AUTHENTICATION BEFORE ANY ACTION 30
FIA_UID.2 USER IDENTIFICATION BEFORE ANY ACTION 31
6.1.3 Class FDP – Access control policy 31
FDP_ACC.1 SUBSET ACCESS CONTROL 31
FDP_ACF.1 SECURITY ATTRIBUTE BASED ACCESS CONTROL 31
6.1.4 Class FMT – Security Management 31
FMT_MSA.1 MANAGEMENT OF SECURITY ATTRIBUTES 31
FMT_MSA.3 STATIC ATTRIBUTE INITIALISATION 32
FMT_MTD.1 MANAGEMENT OF TSF DATA 32
FMT_SMF.1 SPECIFICATION OF MANAGEMENT FUNCTIONS 36
FMT_SMR.1 SECURITY ROLES 36
6.1.5 Class EXT_WAF – Web Application Firewall 36
6.2 Security assurance requirements 37
6.3 Security requirement rationale 38
6.3.1 Rational for the security functional requirements 38
6.3.2 Dependencies of security functional requirements 41
6.3.3 Rational for the assurance requirements 42
7 TOE SUMMARY SPECIFICATION 43
7.1 SF1 – Security Audit 43
7.2 SF2 – Access Control 44
7.3 SF3 – Security Management 44
7.4 SF4 – Web Application Firewall 46
7.4.1 Data Collection 46
7.4.2 Analysis 47
7.4.3 Reaction 49
7.4.4 Restricted Data Review 49
7.5 SF5 – Identification and Authentication 50
7.6 Rationale on TOE specification 51
8 APPENDIX 53
8.1 References 53
8.2 Acronyms 53

List of Tables

Page

Table 1.1 – ST Identification 6
Table 1.2 – TOE Identification 6
Table 3.1 – Subjects 16
Table 3.2 – Threats 17
Table 3.3 – Organizational Security Policies 17
Table 3.4 – Assumptions 18
Table 4.1 – Security Objectives for the TOE 19
Table 4.2 – Security Objectives for the operational environment 20
Table 4.3 – Security Objectives rationale 20
Table 6.1 – TOE Security Functional Requirements 27
Table 6.2 – Audit Events 28
Table 6.3 – Management Functions 32
Table 6.5 – EAL Security Assurance Requirements 37
Table 6.6 – Fulfillment of Security Objectives 38
Table 6.7 – Dependencies of security requirements 41
Table 7.1 – Sections of a generated report 49
Table 7.2 – Security Requirements vs. Security Functions 51

List of Figures

Page

Figure 1.1 – Sniffer Mode 7
Figure 1.2 – Reverse Proxy Mode 7
Figure 1.3 – Transparent Proxy Mode 8
Figure 1.4 – Bridge Mode 8
Figure 1.5 – TOE Boundary .... **Ошибка! Закладка не определена.**
Figure 5.1 – EXT_WAF: Web Application Firewall class decomposition 23

1 ST Introduction

This chapter presents ST and TOE identification information, summarizes the ST in narrative form and provides information for a potential user to determine whether PT Application Firewall is of interest. A ST contains the security requirements of an identified Target of Evaluation (TOE) and specifies the functional and assurance security measures offered by that TOE to meet stated requirements. A ST principally defines:

a) A security problem expressed as a set of assumptions about the security aspects of the operational environment, a list of threats that the TOE is intended to counter, and any known rules with which the TOE must comply (chapter 3).
b) A set of security objectives and a set of security requirements to address the security problem (chapters 4 and 6).
c) The security functionality provided by the TOE that meets the set of requirements (chapter 7).

1.1 Security Target and TOE references

Table 1.1 – ST Identification

Title:	PT Application Firewall Common Criteria Certification Security Target
Short Title:	PT AF ST
Version:	2.9
Date:	2020-02-18
Author:	Egor Sorokin, Positive Technologies

Table 1.2 – TOE Identification

TOE Identification:	PT Application Firewall
TOE Short:	PT AF
TOE Version:	3.6.3.758
TOE Developer:	Positive Technologies
CC Identification:	Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 5, April 2017 ([CC])
Evaluation Assurance Level:	EAL2 augmented with Basic Flaw Remediation (ALC_FLR. 2).
PP Conformance:	none

1.2 TOE overview

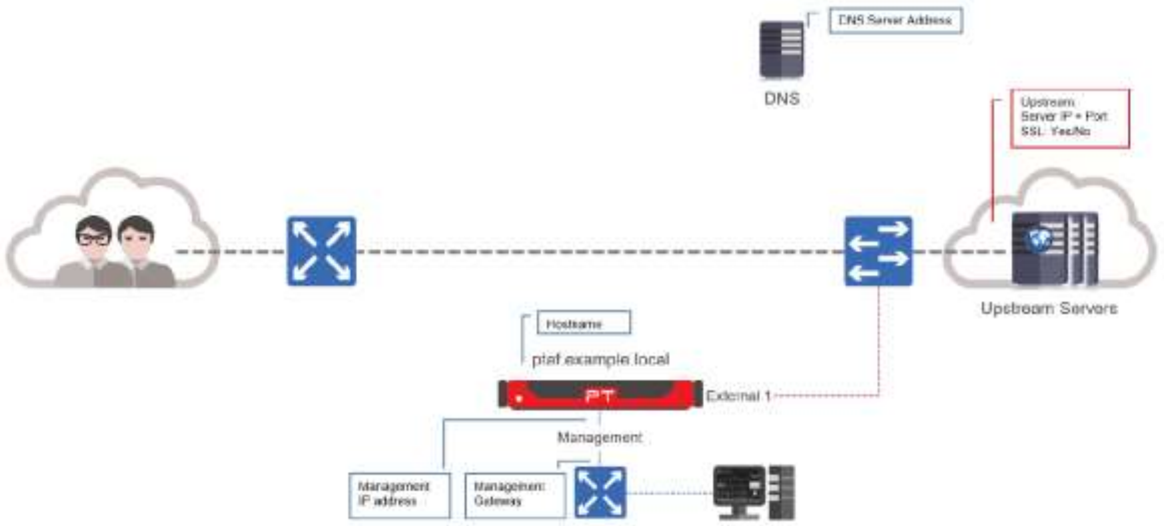
The TOE overview summarizes the usage and major security features of the TOE. The TOE overview provides a context for the TOE evaluation by describing the product and defining the specific evaluated configuration.

The PT Application Firewall (TOE) is a self-learning web application firewall designed to reduce the risks of application attacks if they occur. PT AF applies algorithms to analyze the traffic specifics and the activity of the users who use the applications. Information about the standard user activity is applied to detect potential attacks and deviations in typical user behavior.

The TOE can manage network traffic in multiple modes:

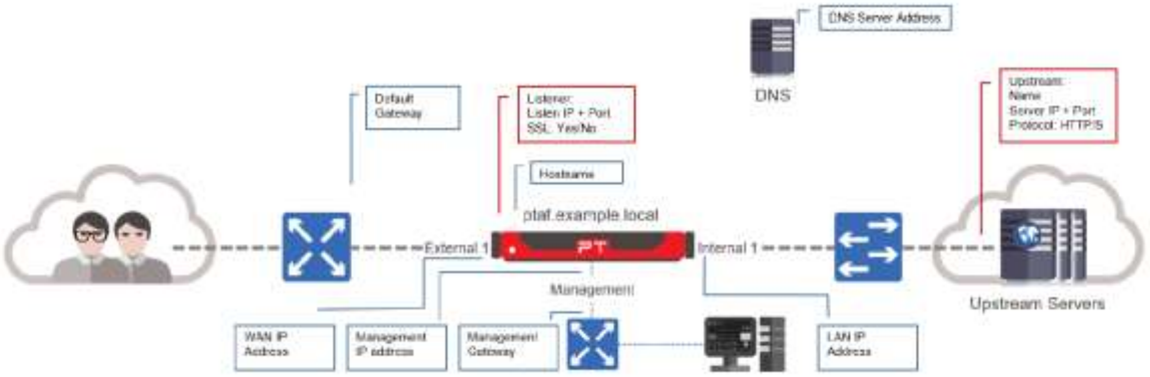
- Sniffer mode.

Figure 1.1 – Sniffer Mode



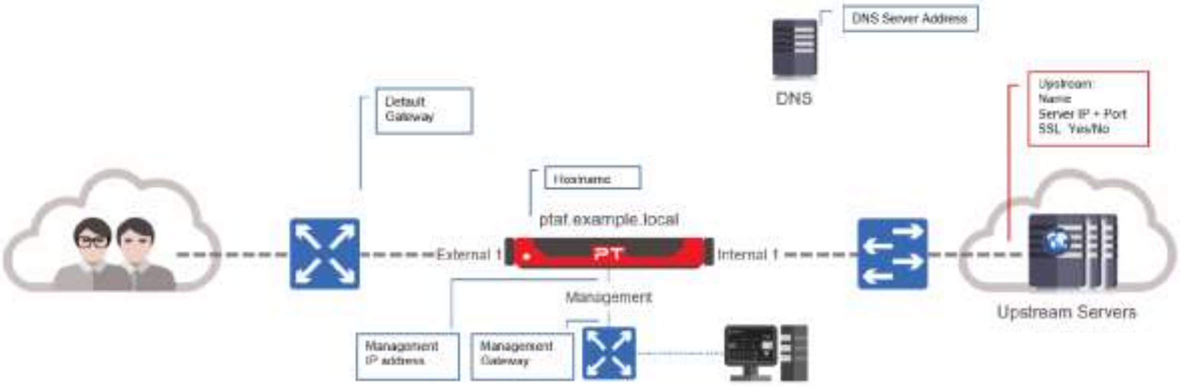
- The reverse proxy mode.

Figure 1.2 – Reverse Proxy Mode



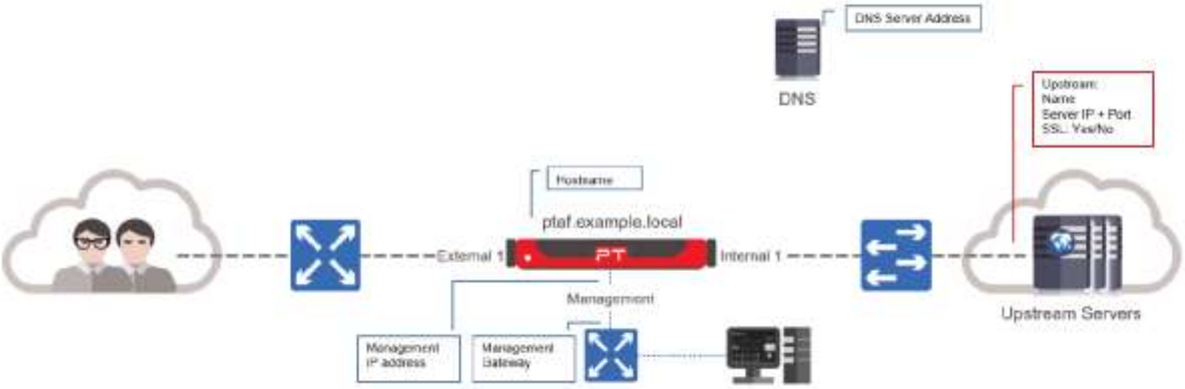
- The transparent proxy mode.

Figure 1.3 – Transparent Proxy Mode



- The bridge mode.

Figure 1.4 – Bridge Mode



In Sniffer Mode, traffic is analyzed without blocking requests and preventing attacks.

In Reverse Proxy Mode, analysis of all requests to a Web application with the possibility of active prevention attacks is performed. The mode is designed to protect the Web application as much as possible: requests are possible only after processing them.

In Transparent Proxy Mode, all HTTP requests to the web application are analyzed with the ability to actively prevent attacks in transparent mode. The TOE operation in this mode is similar to the reverse proxy mode, but does not require changes in the settings of the protected application and network infrastructure.

Bridge Mode is designed to detect intrusions without interfering with the operation of applications. When connecting to the bridge scheme, the TCP session does not break, so the TOE does not affect the functioning of the protected application

1.2.1 Product Type

The PT Application Firewall is designed to monitor and prevent attacks on web applications. The PT AF has been developed as a Debian OS application and its features are compatible with the standard Debian security tools. The TOE consists of software only.

1.2.2 TOE Major Security Features

The TOE major security features are the following:

- Access Control - in the TOE a role-based access control is realized based on which access to certain actions is performed in the User Interface: different permissions can be assigned to each role, while the number of permissions assigned to a particular role is not limited.
- Attack Detection and Prevention – the TOE provides detection of attacks aimed to the protected web applications by different methods: signature (by matching query and response data with known attack signatures), statistical (by building an application or user behavior model and detecting abnormal deviations from the model) and proactive (by using a prior knowledge of attacks and attackers to reduce the likelihood of an attack). The TOE analyses HTTP and HTTPS, XML, XML-based (SOAP and other), JSON, AMF and GWT protocols.
- Auditing – the TOE generates and stores audit records: records about attacks on web applications, records about user actions in the User Interface and records about the status of TOE services. Each record, as a minimum, contains date and time of the event and the event message.
- Identification and Authentication – the TOE requires users to be successfully identified and authenticated before they can access to the user interfaces that are not protected by assumption. The account associates the user’s identity with the user’s password and assigned role.
- Security Functionality Management – the authorized administrators can manage the security functions via the web-based GUI (User Interface).

The user can interact with the TOE using the following capabilities:

- User Interface – is used to display information on attacks and security events as well as to populate databases that are used for TOE configuration,
- WSC shell – is a shell in Debian Operating System for managing TOE network interface configuration,
- REST API – divide into WAF API and WSC API. WSC API supports customization and receives information similar to WSC shell. WAF API allows viewing and installing basic TOE settings that are configurable from the user interface as well.

1.2.3 NON-TOE Hardware/Software/Firmware

Figure 1.5 shows the immediate NON-TOE software of the TOE:

- **Debian 7.8 “Wheezy”** is the operating system which hosts the TOE. The TOE uses several resources of the OS. These resources comprise general functionality as well as specific functionality of the OS, which is necessary for the security functionality of the TOE (e.g. reliable timestamps).
- **MongoDB (Incident Database)** is the database, which is used by the TOE to store the information about configuration and rules. The database has to be installed on the same machine as the TOE itself. The database installation package is included into the PT AF virtual machine.
- **ElasticSearch (Configuration Database)** is the database which is used by the TOE to store the information about security events and alerts. The database has to be installed on the same machine as the TOE itself. The database installation package is included into the TOE virtual machine.

OS Services includes these parts:

- **External Service (syslog)** allows configuring and viewing the configuration of the filtering service and the syslog service. In addition, the syslog section allows setting the severity of events that are to be sent, configuring a protocol and a syslog server IP address, and enabling/disabling messaging.
- **External Service (monit)** allows monitor other TOE services as well as to run and stop them.
- **Iptables** provides packet filtering, network address translation.
- **Update Service –** The waf-scm utility downloads updates from the Positive Technologies server and updates TOE.

All of the listed components are included in the virtual machine supplied with the TOE delivery, but they are not part of the TOE.

To get access to User Interface user has to get one of listed web browsers:

- Microsoft Edge 40 or higher,
- Google Chrome 38 or higher,
- Mozilla Firefox 33 or higher,
- Opera 12 or higher.

For requests to the REST API, user may use any REST client:

- CLI, for example pre-installed Curl,
- Add-ons to the browser, for example browser plug-ins like HTTPRequester or applications like Burp Suite.

Table 1.3 contains minimum requirements to configure a TOE virtual machine.

Table 1.3: TOE virtual machine requirements

Aspect	Minimum parameter values	Recommended parameter values
Hypervisor parameter values		

Aspect	Minimum parameter values	Recommended parameter values
Hypervisor	ESX/ESXi 4.x/5.0/5.1/5.5/6.0	ESX/ESXi 4.x/5.0/5.1/5.5/6.0
Hypervisor physical CPU	A CPU that supports Intel- VT and AMD-V	Xeon E5 or similar CPU is advised
Network ports	Hypervisor network cards are used	Hypervisor network cards are used
Virtual machine parameter values		
Virtual CPU	4 vCPU	12 vCPU
RAM	16 GB	64 GB
Hard disks	300 GB of virtual disk space is advised	1 TB of virtual disk space is advised

1.3 TOE description

This chapter provides context for the TOE evaluation by identifying the product type and describing the evaluated configuration. The main purpose of this chapter is to bind the TOE in physical and logical terms. The chapter starts with a description of the physical scope, the architecture and the logical scope of the TOE.

1.3.1 Physical scope

The TOE is a software product and is delivered to the end user already installed and secure configured for its use as virtual appliance (.ova): PT.AF.3.6.3.758.ova with SHA256 checksum:

20a7ec952a073adc5d7de4f4969758899c69888a4dd4d79f577efa7e2961f5b7

The virtual machine has preinstalled components of the system software and TOE software.

The version of TOE is 3.6.3.758.

In addition the following guidance pdf-documents are part of the TOE:

- PT Application Firewall – Administrator Guide: PT_AG_1.8.pdf with SHA256 checksum: *dd5851d405bacb3a1726955732a10ebaa477c006c5b72c362618d4dc4ce83b00*
- PT Application Firewall – Quick Start Guide: PT_QSG_1.5.pdf with SHA256 checksum: *a858e5f7cbfdfc28df96b0cf3bd62dbb80e90e2498cf602e846f6abc46fa294b*
- PT Application Firewall – Guidance Addendum: PT_AGD_2.0.pdf with SHA256 checksum: *ce5bf92baf20656e483c8a67f750de0de29bb737b0bad6c88033d4aa37d94b41*

All deliverables are placed in a zip archive and stored on a Positive Technologies file server. To download this zip archive user will receive an e-mail from Positive Technologies' official e-mail server (license@ptsecurity.com). The e-mail contains a link to the Positive Technologies file server. The download of the zip archive from the Positive Technologies file server is secured by an SSL connection with a one-side server authentication.

Further, a notification letter that contains SHA256 checksums for every file of the downloadable archive is sent to the respective customer separately via e-mail.

1.3.2 Logical scope and boundary

This section outlines the boundaries of the security functionality of the TOE; the logical boundary of the TOE includes the security functionality described in the following sections.

1.3.2.1 Security Audit

The TOE records the performed management actions on the TOE as well as each identification and authentication attempt. In addition, TOE provides view selected audit events. Only authorized administrators are allowed to read the audit data.

A further purpose of the audit functionality is to hold users accountable for any actions they perform regarding the configuration of TOE Security Functions.

1.3.2.2 Identification and Authentication

The TOE provides a mechanism for identification and authentication of users who communicates with the TOE through the interfaces that are not protected by assumptions. All authorized TOE users must have an account with security attributes that control the user's access to TSF data and management functions.

1.3.2.3 Security Management

The TOE provides a set of functionalities to manage the security functions, configuration, and other features of the TOE components by authorized user roles. To manage the TOE authorized users have to use the graphical User Interface.

1.3.2.4 Security Access

This function of the TOE controls the access of user groups to data protected by the TOE, whereby it uses an access control policy based on roles. It further controls that only authorized user roles can manage the TOE. Especially the access control policy ensures that only administrators are able to create user accounts and user groups and set privileges to them.

1.3.2.5 Web Application Firewall

The TOE analyzes the network traffic at an application level. Traffic processed by PT AF goes through a set of protectors.

The TOE uses the following types of analysis for detection of attacks aimed to the protected applications:

- Signature,
- Statistical,
- Proactive.

1.3.3 TOE Evaluated Configuration

The TOE evaluated configuration is described as follows:

- A single instance of the TOE (including Debian 7.8, Mongo DB and ElasticSearch) is configured in the local network. There are cluster configurations, but they don't belong to the evaluated TOE configuration,
- Four modes of operation have been tested in the evaluated configuration:
  - Sniffer mode;
  - Reverse Proxy mode;
  - Transparent Proxy mode;
  - Bridge mode.
- The TOE might be deploying to a physical gateway, but this has not been tested in evaluated configuration. The evaluated configuration belongs only to a virtual environment,
- Registration of users in the TOE is carried out using the web interface, introducing their data in the interface available in the User Interface. The TOE supports retrieving users from an Active Directory, but this don't belong to the evaluated TOE configuration,
- The TOE configuration is performed in a way that only those TOE interfaces required for deploying the TSF are configured. This includes the following features available in the TOE are not evaluated (default configuration is kept):
  - Send alert information via SMTP/SMTPS,
  - Sent alert information via SNMPv3,
  - ICAP service,
  - LDAP service,
  - Integration with external systems or products,
  - The Blackbox scanner.
- In the TOE evaluated configuration, for obtaining reliable time sources, the TOE Virtual Machine is configured so that it synchronizes its system clock with the clock of the host machine, instead of using an external secure NTP server,
- The TOE evaluated configuration has been deployed and tested using a hardware and software described below:
  - Burp Suite 1.7.27,
  - Firefox 52, Chrome 63,
  - ESXi 6.0, Intel Xeon E5,
  - 10vCPU, 24GB RAM, 300GB hard disk.

In order to achieve the described configuration, the TOE preparative guides must be thoroughly followed for the TOE installation and configuration.

1.4 Conventions

The CC allows for assignment, refinement, selection and iteration operations to be performed on security functional requirements. All of these operations are used within this ST. These operations are performed as described in Part 2 of the CC, and are shown as follows:

- Completed assignment statements are identified using [*italicized text within brackets*].
- Completed selection statements are identified using [underlined text within brackets].
- Refinements are identified using **bold text**. Any text removed is stricken (Example: ~~TSF Data~~) and should be considered as a refinement.
- Extended Functional and Assurance Requirements are identified using "EXT_" at the beginning of the short name.
- Iterations are identified by appending a number in parentheses following the component title. For example, FAU_GEN.1 (1) Audit Data Generation would be the first iteration and FAU_GEN.1 (2) Audit Data Generation would be the second iteration.

2 Conformance Claims

This section provides the identification for any CC, Protection Profile (PP), and EAL package conformance claims. Rationale is provided for any extensions or augmentations to the conformance claims. This chapter contains the following sections:

- CC conformance claims
- PP claim
- Package claim
- Conformance rationale

2.1 CC conformance claims

This Security Target claims to be conformant to the Common Criteria 3.1R5:

- Part 2 extended to [CC]

  In order to provide a complete description of the functional requirements addressed by the TOE, functional components of part 2 of the Common Criteria framework were used. But also additions to the Common Criteria part 2 were defined, to fulfill the requirement of a complete and consistent TOE description.

- Part 3 conformant to [CC]

  For the description of the requirements due to the trustworthiness of the TOE, only security assurance requirements of CC part 3 were used.

2.2 PP claim

This ST does not claim conformance to any PP.

2.3 Package claim

This Security Target claims to be conformant to the Security Assurance Requirements package EAL 2 augmented with Basic Flaw Remediation (ALC_FLR. 2).

2.4 Conformance rationale

This ST does not claim conformance to any PP.

3 Security Problem Definition

This section describes the security aspects of the operational environment in which the TOE will be used and the manner in which the TOE is expected to be employed. It describes

- the assets that have to be protected by the TOE,
- threats against those assets,
- organizational security policies that TOE shall comply with, and
- assumptions about the operational environment of the TOE.

3.1 Assets

All data from and to the interfaces available on the TOE is categorized into TSF data and non-TSF data. The *integrity* and *confidentiality* of all this data is to be protected and considered as assets.

TSF data:

- User account data (user identifier, user role, user password),
- Audit records (attack event log, user actions, service event log),
- Configuration data of security features and functions.

Non-TSF data:

- Traffic and contents stored in the web applications protected by the TOE.

3.2 Subjects

The following table lists all threat agents that interact with the TOE. The threat agents are divided into two categories:

Table 3.1 – Subjects

Subject	Description
Attackers (who are not TOE users)	They have public knowledge of how the TOE operates and are assumed to possess a basic skill level, limited resources to alter TOE configuration settings or parameters.
TOE users	They have extensive knowledge of how the TOE operates and are assumed to possess a high skill level, moderate resources to alter TOE configuration settings or parameters. TOE users try to manipulate configuration data that they are not authorized to access and try to manipulate the safe operation of the TOE.

3.3 Threats

The table below identifies the threats to the assets against which protection is required by the TOE:

Table 3.2 – Threats

Threat	Description
T.CONFIG	The TOE may be configured in such a way that Attacker and TOE user may gain unauthorized access to TSF data
T.UNAUTH	An attacker and TOE user may gain unauthorized access to the TOE and disclose TSF data and/or modify the behavior of the TOE in an unsecure way.
T.UNDETECTED	The security relevant actions of users may go undetected making the TSF data vulnerable to attack. An attacker could exploit these circumstances to get unauthorized access or to modify TSF data.

3.4 Organizational security policies

An Organizational Security Policy (OSP) is a set of security rules, procedures, or guidelines imposed by an organization on the operational environment of the TOE.

The following table describes the organizational security policies relevant to the operation of the TOE.

Table 3.3 – Organizational Security Policies

Policy	Description
P.PATTERN	Files and data on protected machines have to be checked for known patterns.
P.ACCESSCONTROL	The TOE provides a set of user roles with different privileges to perform certain actions. Not all the users are authorized to perform all the possible actions.

3.5 Assumptions

This section describes the security aspects of the intended operational environment for the evaluated TOE. The operational environment must be managed in accordance with assurance requirement documentation for delivery, operation, and user guidance. The following specific conditions are required to ensure the security of the TOE and are assumed to exist in an environment where this TOE is employed.

The assumptions about the TOE’s security environment are defined in Table 3.4 below.

Table 3.4 – Assumptions

Assumption	Description
A.AUTHEN	The underlying OS provides an identification and authentication mechanism to protect WSC interface and TSF data.
A.MANAGE	The users who manage the TOE as well as all administrators of the host, where the TOE is installed on, are non-hostile, appropriately trained, and follow all guidance documentation.
A.PROTECT	The TOE is located within controlled access facilities that will be protected from unauthorized physical access and modification.
A.STORAGE	The operational environment provides the availability to store data within an external database. The database is located at the same machine as the TOE itself and it is ensured that only authorized users are allowed to access the TSF data stored within this database.
A.OS	The underlying OS provides the TOE with the necessary reliable timestamps and authentication in those interfaces for which the authentication is not provided by the TOE.

4 Security Objectives

Security objectives are concise, abstract statements of the intended solution to the problem defined by the security problem definition (see chapter 3). The set of security objectives for a TOE form a high-level solution to the security problem. This high-level solution is divided into two part-wise solutions: the security objectives for the TOE, and the security objectives for the TOE"s operational environment. This section identifies the security objectives for the TOE and its supporting environment.

4.1 Security objectives for the TOE

TOE security objectives are defined in Table 4.1, below.

Table 4.1 – Security Objectives for the TOE

Objective	Description
O.ACCESS	The TOE must enforce an access control policy to ensure that only users who are authorized to access the relevant data are able to do so.
O.AUDIT	The TOE must provide the means of recording the performed management actions on the TOE as well as each identification and authentication attempt, so as to assist corresponding users in the detection of potential attacks or misconfiguration of TOE security features as well as hold users accountable for any actions they perform regarding the configuration of TOE Security Functions.
O.AUTH	The TOE must provide a mechanism for identification and authentication of users who communicate with the TOE via User Interface.
O.MANAGE	The TOE must include a set of functions that allow efficient management of its functions and data, ensuring that TOE users with the appropriate privileges, and only those TOE users, may exercise such control.
O.RESPONSE	The TOE must respond to attacks against the web systems it identifies based on its configuration
O.ANALYSIS	The TOE must analyze requests that come to the web systems in order to identify attacks against the protected web systems, must be able to record the results of its analysis and must collect information about all events that are indicative of inappropriate activity that may have resulted from misuse, access, or malicious activity directed to protected web systems
O.PATTERN	The TOE must provide mechanisms to detect and take action against known patterns introduced to the protected computer via network traffic.

4.2 Security objectives for the operational environment

The security objectives for the TOE operational environment are based on the secure usage assumptions and defined in Table 4.2 below.

Table 4.2 – Security Objectives for the operational environment

Objective	Description
OE.AUTHEN	The underlying OS must provide an identification and authentication mechanism to protect WSC interface and TSF data.
OE.MANAGE	TOE users who are responsible for the TOE must be competent, non-hostile, appropriately trained and follow all guidance documentation.
OE.PROTECT	Those responsible for the TOE must ensure that the physical environment must be suitable for supporting a computing device in a secure setting.
OE.STORAGE	The operational environment must provide the availability to store data within an external DB. The DB must be located at the same machine as the TOE itself.
OE.OS	The underlying OS must provide reliable timestamps and external authentication services.

4.3 Security objectives rationale

Table 4.3 – Security Objectives rationale

Threats, OSPs and Assumptions vs. Security Objectives	O.ACCESS	O.AUDIT	O.AUTH	O.MANAGE	O.RESPONSE	O.ANALYSIS	O.PATTERN	OE.AUTHEN	OE.MANAGE	OE.PROTECT	OE.STORAGE	OE.OS
T.CONFIG				X								
T.UNAUTH		X	X	X								
T.UNDETECTED		X										
P.ACCESSCONTROL	X											
P.PATTERN					X	X	X					
A.AUTHEN								X				
A.MANAGE									X			
A.PROTECT										X		

Threats, OSPs and Assumptions vs. Security Objectives	O.ACCESS	O.AUDIT	O.AUTH	O.MANAGE	O.RESPONSE	O.ANALYSIS	O.PATTERN	OE.AUTHEN	OE.MANAGE	OE.PROTECT	OE.STORAGE	OE.OS
A.STORAGE											X	
A.OS												X

T.CONFIG is countered by

- O.MANAGE since this TOE security objective ensures that the TOE provides a set of management functions that allow a secure configuration of the TOE. Further this objective ensures that only users with appropriate privileges are able to manage the TOE.

T.UNAUTH is countered by a combination of

- O.MANAGE since this TOE security objective ensures that only users with appropriate privileges are able to manage the TOE, and
- O.AUTH since this TOE security objective provides an authentication and identification mechanism.
- O.AUDIT since this TOE security objective ensures that the performed management actions on the TOE as well as each identification and authentication attempt are recorded.

T.UNDETECTED is countered by a combination of

- O.AUDIT since this TOE security objective ensures that the performed management actions on the TOE as well as each identification and authentication attempt are recorded.

As can be seen above every identified threat is countered by one or more security objectives as defined in Table 4.1.

P.PATTERN is fulfilled by

- O. ANALYSIS – this TOE security objective ensures that the TOE is able to analyze collected WAF data in order to identify misuse and unauthorized or malicious activity in the protected web systems, and be able to record the results of its analysis.
- O.RESPONSE – supports O. ANALYSIS in addressing this threat by ensuring the TOE is able to respond to identified misuse and unauthorized or malicious activity.
- O.PATTERN - makes sure that data and files on protected machines are checked for known patterns attacks to implement this organization policy.

P.ACCESSCONTROL is fulfilled by

- O.ACCESS since this TOE security objective ensures that an access control policy is applied, which ensures that only users who are authorized to access sensitive data are able to do so.

Every organizational security policy is fulfilled by one or more security objectives.

A.AUTHEN is addressed by

- OE.AUTHEN, since OE.AUTHEN ensures that the underlying OS provides an identification and authentication mechanism for the TOE.

A.MANAGE is addressed by

- OE.MANAGE, since OE.MANAGE ensures that users who manage the TOE are non-hostile, appropriately trained, and follow all guidance documentation.

A.PROTECT is addressed by

- OE.PROTECT, since OE.PROTECT ensures that the TOE operational environment provides protection from external interference or tampering.

A.STORAGE is addressed by

- OE.STORAGE, since OE.STORAGE ensures that the TOE operational environment provides the availability to store data within an external DB that is located at the same machine as the TOE itself.

A.OS is addressed by

- OE.OS, since OE.OS ensures that the underlying OS provides reliable timestamps to the TOE and external authentication services.

Every assumption is addressed by one objectives for the operational environment. The justification above demonstrates that the defined security objectives for the operational environment uphold all defined assumptions.

5 Extended Components Definition

This chapter defines TOE security functional requirements and assurance requirements which are not part of CC 3.1 part 2 or part 3.

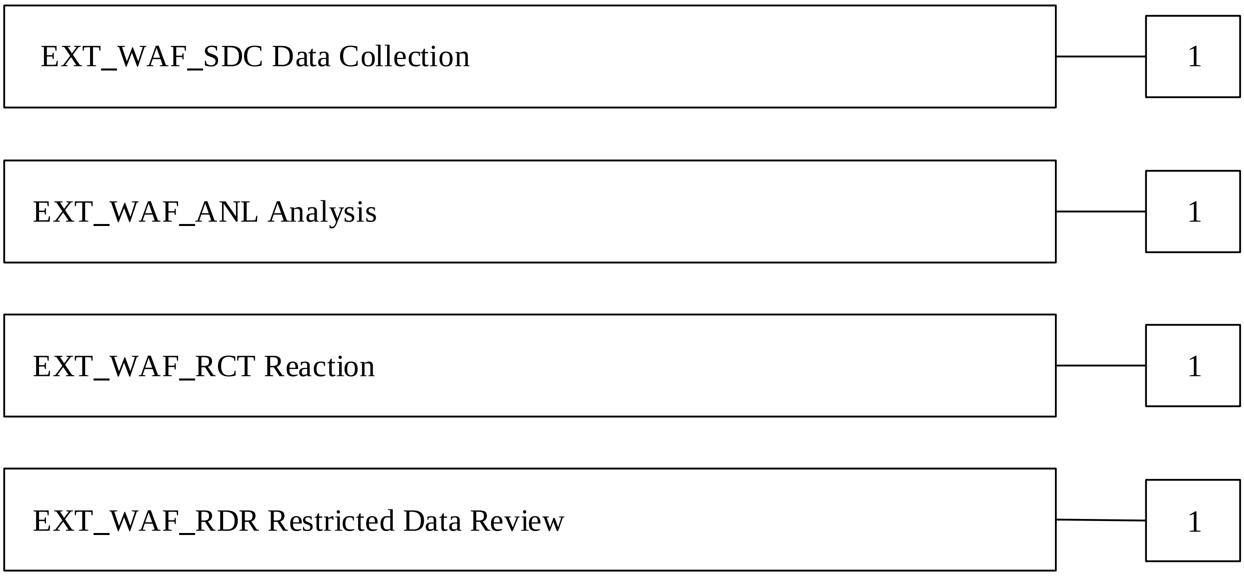
5.1 Extended TOE Security Functional Components

This section specifies the extended SFRs for the TOE.

5.1.1 Class EXT_WAF: Web Application Firewall

This ST defines a new functional class for use within this ST: Web Application Firewall (WAF). This family of WAF requirements was created to specifically address the data collected and analyzed by a WAF. The audit family of the CC (FAU) was used as a model for creating these requirements. The purpose of this family of requirements is to address the unique nature of WAF data and specify requirements for collecting, analyzing and reviewing WAF data.

Figure 5.1 – EXT_WAF: Web Application Firewall class decomposition


EXT_WAF_SDC Data Collection
1
EXT_WAF_ANL Analysis
1
EXT_WAF_RCT Reaction
1
EXT_WAF_RDR Restricted Data Review
1

5.1.1.1 (EXT_WAF_SDC) Data Collection

Family Behavior

This family defines requirements for being able to collect WAF data from targeted web server/web application. This family defines a minimum set of information to be collected and recorded.

Component leveling

EXT_WAF_SDC Data Collection
1

EXT_WAF_SDC.1 System Data Collection provides for the functionality to require TSF collection of data that may be related to security events.

Management: EXT_WAF_SDC.1

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

a) Maintenance of the parameters that control data collection.

Audit: EXT_WAF_SDC.1

There are no auditable events foreseen.

EXT_WAF_SDC.1 Data Collection

Hierarchical to: No other components.

Dependencies: No other dependencies.

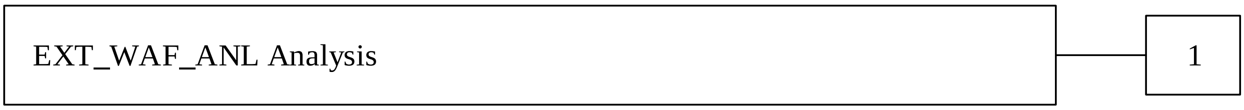
EXT_WAF_SDC.1.1 TSF shall be able to collect the following information from the web systems: [assignment: specifically defined information].

5.1.1.2 (EXT_WAF_ANL) Analysis

Family Behaviour

This family defines the requirements for the TOE regarding analysis of information related to security events.

Component leveling



EXT_WAF_ANL.1 Analysis provides for the functionality to require TSF controlled analysis of data collected that is related to security events.

Management: EXT_WAF_ANL

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

b) Maintenance of the parameters that control data analysis.

Audit: EXT_WAF_ANL

There are no auditable events foreseen.

EXT_WAF_ANL.1 Analysis

Hierarchical to: No other components.

Dependencies: FPT_STM.1 Reliable time stamps & EXT_WAF_SDC.1 System Data Collection.

EXT_WAF_ANL.1.1 The TSF shall perform the following analysis function(s) on the collected data: [assignment: analytical functions].

EXT_WAF_ANL.1.2 The TSF shall record within each analytical result at least the date and time of the event and the following information: [assignment: information that shall be recorded]

5.1.1.3 (EXT_WAF_RCT) Reaction

Family Behaviour

This family defines the requirements for the TOE regarding reactions to the analysis of information related to security events received from web system when an attack is detected.

Component leveling

EXT_WAF_RCT.1 Reaction provides for the functionality to require TSF controlled reaction to the analysis of data received from protected web systems regarding information related to security events when an attack is detected.

Management: EXT_WAF_RCT.1

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

c) Maintenance of the parameters that control reaction.

Audit: EXT_WAF_RCT.1

There are no auditable events foreseen.

EXT_WAF_RCT.1 Reaction

Hierarchical to: No other components.

Dependencies: EXR_WAF_ANL.1

EXT_WAF_RCT.1.1 The TSF shall send an alarm to [assignment: alarm destination] and take [assignment: appropriate actions] when an attack is detected.

5.1.1.4 (EXT_WAF_RDR) Restricted data review

Family Behaviour

This family defines the requirements for system data tools that should be available to authorized users to assist in the review of system data.

Component leveling

EXT_WAF_RDR.1 Restricted data review, the TSF shall provide the System data in an understandable form only to authorized users.

Management: EXT_WAF_RCT.1

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

d) Maintenance of the group of users with read access rights to collected data.

Audit: EXT_WAF_RDR.1

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

a) All report generation attempts.

EXT_WAF_RDR.1 Restricted data review

Hierarchical to: No other components.

Dependencies: EXT_WAF_SDC.1, EXT_WAF_ANL.1, FMT_SMR.1

EXT_WAF_RDR.1.1 The TSF shall provide [assignment: authorized users] with the capability to read [assignment: list of data].

EXT_WAF_RDR.1.2 The TSF shall provide the collected data in a manner suitable for the user to interpret the information.

EXT_WAF_RDR.1.3 The TSF shall prohibit all users read access to the collected data, except those users that have been granted explicit read-access

5.2 Extended TOE Security Assurance Components

There are no extended TOE Security Assurance Components.

6 Security Requirements

This chapter defines the IT security requirements that shall be satisfied by the TOE or its operational environment:

Common Criteria divides TOE security requirements into two categories:

- Security functional requirements (SFRs) (such as, identification and authentication, security management, and user data protection) that the TOE and the supporting evidence need to satisfy to meet the security objectives of the TOE.
- Security assurance requirements (SARs) that provide grounds for confidence that the TOE and its supporting IT environment meet its security objectives (e.g. configuration management, testing, and vulnerability assessment).

These requirements are discussed separately within the following subchapters.

6.1 Security functional requirements

The specified functional requirements are compliant with Common Criteria v3.1 part 2 and are corresponding with the given functional components.

Table 6.1 – TOE Security Functional Requirements

Name	Description
FAU_GEN.1	Audit Data Generation
FAU_GEN.2	User identity association
FAU_SAR.1	Audit review
FAU_SAR.2	Restricted audit review
FIA_ATD.1	User attribute definition
FIA_UAU.2	User authentication before any action
FIA_UID.2	User identification before any action
FDP_ACC.1	Subset access control
FDP_ACF.1	Security attribute based access control
FMT_MSA.1	Management of security attributes
FMT_MSA.3	Static attribute initialisation
FMT_MTD.1	Management of TSF data
FMT_SMF.1	Specification of management functions
FMT_SMR.1	Security roles
EXT_WAF_SDC.1	Data Collection
EXT_WAF_ANL.1	Analysis
EXT_WAF_RCT.1	Reaction
EXT_WAF_RDR.1	Restricted Data Review

6.1.1 Class FAU – Security Audit

FAU_GEN.1 Audit Data Generation

Hierarchical to: No other components.

Dependencies: FPT_STM.1 Reliable time stamps

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 audit functions;

b) All auditable events, for the [not specified] level of audit; and

c) [*identification and authentication attempts (success and failure) as well as all actions that are listed in Table 6.2].*

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 (if applicable), and the 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, [*IP-address*].

Table 6.2 – Audit Events

Objects	Operation
Dashboard	exclude
Policies	create, edit, delete
Web Applications	create, edit, delete
Services	create, edit, delete
SSL files	upload, delete
SSL configurations	create, edit, delete
Rules and security events	create, edit, delete
Actions	create, edit, delete
Tags	create, edit, delete
HMM	switch, clear, train
Content Security Policy	create, edit, delete
IP Blacklist	create, edit, delete
Host Blacklist	create, edit, delete
XML Schemas	upload, delete
Form Policies	create, edit, delete
Web UI settings	edit
Web UI Security settings	edit
Regular expression types	create, edit, delete

Objects	Operation
Gateways	edit
Upstreams	create, edit, delete
Firewall	create, edit, delete
Network interface aliases	create, edit, delete
Virtual IP addresses	create, edit, delete
Source IP ranges	create, edit, delete
Events	create, edit, delete
Alerts	create, edit, delete
ICAP services	create, edit, delete
LDAP services	create, edit, delete
Suspicious sessions	edit, delete
Users	create, edit, delete
User groups	create, edit, delete
Active Directory settings	create, edit, delete
Troubleshooting	create
Forensics	upload, delete
Reports	delete
Report templates	create, edit, delete
Report schedule	create, edit, delete
Backups	create, delete, upload, restore
Backup schedule	create, edit, delete
Virtual patching	delete

FAU_GEN.2 User identity association

Hierarchical to: No other components.

Dependencies: FAU_GEN.1 Audit data generation

FIA_UID.1 Timing of identification

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.

FAU_SAR.1 Audit review

Hierarchical to: No other components.

Dependencies: FAU_GEN.1 Audit data generation

FAU_SAR.1.1 The TSF shall provide [*Administrators*] with the capability to read [*all information*] 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.

FAU_SAR.2 Restricted audit review

Hierarchical to: No other components.

Dependencies: FAU_SAR.1 Audit review

FAU_SAR.2.1 The TSF shall prohibit all users read access to the audit records, except those users that have been granted explicit read-access.

6.1.2 Class FIA – Identification and authentication

FIA_ATD.1 User attribute definition

Hierarchical to: No other components.

Dependencies: No dependencies.

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

- *User identifier (Login),*
- *Role (Group membership)*],
- *User Password.*

FIA_UAU.2 User authentication before any action

Hierarchical to: FIA_UAU.1 Timing of authentication

Dependencies: FIA_UID.1 Timing of identification

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.

FIA_UID.2 User identification before any action

Hierarchical to: FIA_UID.1 Timing of identification

Dependencies: No dependencies.

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

6.1.3 Class FDP – Access control policy

FDP_ACC.1 Subset access control

Hierarchical to: No other components.

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACC.1.1 The TSF shall enforce the [*Role-based access control policy*] on [

- *Subjects: all authenticated users*
- *objects as listed in the second column of Table 6.3 and*
- *all operations listed in the first column of Table 6.3*].

FDP_ACF.1 Security attribute based access control

Hierarchical to: No other components.

Dependencies: FDP_ACC.1 Subset access control

FMT_MSA.3 Static attribute initialisation

FDP_ACF.1.1 The TSF shall enforce the [*Role-based access control policy*] to objects based on the following: [

- *The user identity and group membership that is associated with a role*].

FDP_ACF.1.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [

- *If the requested operation on the requested object is permitted to the role that is associated with a group of which the authenticated user is a member (cf. Table 6.3), grant access,*
- *else deny access*].

FDP_ACF.1.3 The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: [*no additional rules*].

FDP_ACF.1.4 The TSF shall explicitly deny access of subjects to objects based on the following additional rules: [*no additional rules*].

6.1.4 Class FMT – Security Management

FMT_MSA.1 Management of security attributes

Hierarchical to: No other components.

Dependencies: FDP_ACC.1 Subset access control

FMT_SMR.1 Security roles

FMT_SMF.1 Specification of Management Functions

FMT_MSA.1.1 The TSF shall enforce the [*Role-based access control policy*] to restrict the ability to [*manage*] the security attributes [*group membership of the user roles*] to [*the authorized user role Administrator*].

FMT_MSA.3 Static attribute initialisation

Hierarchical to: No other components.

Dependencies: FMT_MSA.1 Management of security attributes

FMT_SMR.1 Security roles

FMT_MSA.3.1 The TSF shall enforce the [*Role-based access control policy*] to provide [restrictive] default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2 The TSF shall allow the [*no one*] to specify alternative initial values to override the default values when an object or information is created.

FMT_MTD.1 Management of TSF data

Hierarchical to: No other components.

Dependencies: FMT_SMF.1 Specification of management functions

FMT_SMR.1 Security roles

FMT_MTD.1.1 The TSF shall restrict the ability to [delete, [view, create, edit, and other operations as defined in column «Operation» of table 6.3]] the [*objects as defined in column "Objects" of Table 6.3*] to [*the authorized identified roles as defined in column "Authorized User Roles" of Table 6*.3].

Table 6.3 – Management Functions

Objects	Operation	Authorized User Roles
Dashboard	view, scan, exclude, block	Administrator
	–	Configurator
Autodiscovery Wizard	view	Administrator
	–	Configurator
Policies	view, create, edit, delete	Administrator
	view, create, edit, delete	Configurator
Web Applications	view, create, edit, delete	Administrator
	view, create, edit, delete	Configurator
Services	view, create, edit, delete	Administrator
	view, create, edit, delete	Configurator
SSL files	view, upload, delete	Administrator
	–	Configurator
SSL configurations	view, create, edit, delete	Administrator
	–	Configurator

Objects	Operation	Authorized User Roles
Rules and security events	view, create, edit, copy, delete	Administrator
	–	Configurator
Actions	view, create, edit, delete	Administrator
	–	Configurator
Tags	view, create, edit, delete	Administrator
	–	Configurator
HMM	View, switch, clear, train	Administrator
	–	Configurator
Content Security Policy	view, create, edit, delete	Administrator
	–	Configurator
IP Blacklist	view, create, edit, delete	Administrator
	–	Configurator
Host Blacklist	view, create, edit, delete	Administrator
	–	Configurator
XML Schemas	view, upload, delete	Administrator
	–	Configurator
Form Policies	view, create, edit, delete	Administrator
	–	Configurator
Regular expression types	view, create, edit, delete	Administrator
	–	Configurator
Gateways	view, edit	Administrator
	view, edit	Configurator
Upstreams	view, create, edit, delete	Administrator
	view, create, edit, delete	Configurator
Firewall	view, create, edit, delete, list	Administrator
	–	Configurator
Sniffer	view	Administrator
	–	Configurator
Network interface aliases	view, create, edit, delete	Administrator

Objects	Operation	Authorized User Roles
	view, create, edit, delete	Configurator
Virtual IP addresses	view, create, edit, delete	Administrator
	–	Configurator
Source IP ranges	view, create, edit, delete	Administrator
	view, create, edit, delete	Configurator
Cluster	view	Administrator
	–	Configurator
Events	view, create, edit, delete	Administrator
	–	Configurator
Alerts	view, create, edit, delete	Administrator
	–	Configurator
ICAP services	view, create, edit, delete	Administrator
	–	Configurator
LDAP services	view, create, edit, delete	Administrator
	–	Configurator
Suspicious sessions	view, edit, delete	Administrator
	–	Configurator
Service management	view	Administrator
	–	Configurator
Service event log	view	Administrator
	–	Configurator
Users	view, create, edit, delete	Administrator
	–	Configurator
User groups	view, create, edit, delete	Administrator
	–	Configurator
User actions	view	Administrator
	–	Configurator
Active Directory settings	view, create, edit, delete	Administrator
	–	Configurator
Trainer settings	view	Administrator

Objects	Operation	Authorized User Roles
	–	Configurator
Web UI settings	view, edit	Administrator
	–	Configurator
Web UI security settings	view, edit	Administrator
	–	Configurator
SMTP	view	Administrator
	–	Configurator
Rules update	view	Administrator
	view	Configurator
Remote access	view	Administrator
	–	Configurator
Access log	view	Administrator
	–	Configurator
Commands log	view	Administrator
	–	Configurator
Troubleshooting	view, create	Administrator
	–	Configurator
About	view	Administrator
	–	Configurator
Forensics	view, upload, delete	Administrator
	–	Configurator
Reports	view, download, delete	Administrator
	–	Configurator
Report templates	view, create, edit, delete	Administrator
	–	Configurator
Report schedule	view, create, edit, delete	Administrator
	–	Configurator
Backups	view, create, download, delete, upload, restore	Administrator
	–	Configurator

Objects	Operation	Authorized User Roles
Backup schedule	view, create, edit, delete	Administrator
	–	Configurator
IP Whois	view	Administrator
	–	Configurator
Regular expression tester	view	Administrator
	–	Configurator
Trainer debug	view	Administrator
	–	Configurator
Virtual patching	view, delete	Administrator
	–	Configurator

FMT_SMF.1 Specification of Management Functions

Hierarchical to: No other components.

Dependencies: No Dependencies

FMT_SMF.1.1 The TSF shall be capable of performing the following security management functions [*see Table 6.3*].

FMT_SMR.1 Security roles

Hierarchical to: No other components.

Dependencies: FIA_UID.1 Timing of identification

FMT_SMR.1.1 The TSF shall maintain the roles [*as listed in Table 6.4*].

FMT_SMR.1.2 The TSF shall be able to associate users with roles.

Table 6.4 – TOE Roles

Subject	Description
Administrator	An authenticated user who has unrestricted access to all TOE functionalities. Administrators are responsible for the management of all TOE process and have to ensure that the TOE operates in a secure way. Especially only Administrators are allowed to create, remove and change users, user groups and object owners, but they are not able to view passwords of TOE users.
Configurator	An authorized user who has permissions to change network settings and basic settings of policies/server groups.
Custom	An authorized user who has customizable permissions

6.1.5 Class EXT_WAF – Web Application Firewall

EXT_WAF_SDC.1 Data Collection

Hierarchical to: No other components.

Dependencies: No other dependencies.

EXT_WAF_SDC.1.1 TSF shall be able to collect the following information from the web systems: [*network traffic*]

EXT_WAF_ANL.1 Analysis

Hierarchical to: No other components.

Dependencies: FPT_STM.1 Reliable time stamps & EXT_WAF_SDC.1 System Data Collection.

EXT_WAF_ANL.1.1 The TSF shall perform the following analysis functions on the collected data [*statistical, signature, proactive*].

EXT_WAF_ANL.1.2 The TSF shall record within each analytical result at least the date and time of the event and the following information [*event severity, name of event, event description, event tag, client IP-address, client geolocation, request*].

EXT_WAF_RCT.1 Reaction

Hierarchical to: No other components.

Dependencies: EXT_WAF_ANL.1

EXT_WAF_RCT.1.1 The TSF shall send an alarm to [*attack log*] and take [*action specified by the protector that was triggered by the event*] when an attack is detected.

EXT_WAF_RDR.1 Restricted data review

Hierarchical to: No other components.

Dependencies: EXT_WAF_SDC.1, EXT_WAF_ANL.1, FMT_SMR.1

EXT_WAF_RDR.1.1 The TSF shall provide [*the users associated with the roles administrator*] with the capability to read [*information about all collected attacks*].

EXT_WAF_RDR.1.2 The TSF shall provide the collected data in a manner suitable for the user to interpret the information.

EXT_WAF_RDR.1.3 The TSF shall prohibit all users read access to the collected data, except those users that have been granted explicit read-access.

6.2 Security assurance requirements

The Security Assurance Requirements for the TOE are the assurance components of Evaluation Assurance Level 2 (EAL2) augmented with Basic Flaw Remediation (ALC_FLR. 2). They are all drawn from Part 3 of the Common Criteria. The assurance components are listed in Table 6.5.

Table 6.5 – EAL Security Assurance Requirements

Assurance component	Name
ADV_ARC.1	Security architecture description
ADV_FSP.2	Security-enforcing functional specification
ADV_TDS.1	Basic design
AGD_OPE.1	Operational user guidance
AGD_PRE.1	Preparative procedures

Assurance component	Name
ALC_CMC.2	Use of a CM system
ALC_CMS.2	Parts of the TOE CM coverage
ALC_DEL.1	Delivery procedures
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_COV.1	Evidence of coverage
ATE_FUN.1	Functional testing
ATE_IND.2	Independent testing – sample
AVA_VAN.2	Vulnerability analysis
ALC_FLR.2	Basic Flaw Remediation

6.3 Security requirement rationale

6.3.1 Rational for the security functional requirements

Table 6.6 – Fulfillment of Security Objectives

Security Objectives vs. Security Requirements	O.ACCESS	O.AUDIT	O.AUTH	O.MANAGE	O.RESPONSE	O.ANALYSIS	O.PATTERN
FAU_GEN.1		X					
FAU_GEN.2		X					
FAU_SAR.1		X					
FAU_SAR.2		X					
FIA_ATD.1			X				
FIA_UAU.2			X				
FIA_UID.2			X				
FDP_ACC.1	X						

Security Objectives vs. Security Requirements	**O.ACCESS**	**O.AUDIT**	**O.AUTH**	**O.MANAGE**	**O.RESPONSE**	**O.ANALYSIS**	**O.PATTERN**
FDP_ACF.1	X						
FMT_MSA.1				X			
FMT_MSA.3				X			
FMT_MTD.1				X			
FMT_SMF.1				X			
FMT_SMR.1				X			
EXT_WAF_SDC.1						X	
EXT_WAF_ANL.1						X	X
EXT_WAF_RCT.1					X		X
EXT_WAF_RDR.1		X					X

O.ACCESS

- FDP_ACC.1 – This requirement meets the objective O.ACCESS by ensuring that the access control SFP is enforced by the TOE.
- FDP_ACF.1 – This requirement meets the objective O.ACCESS by ensuring that the rules regarding the access control are applied.

O.AUDIT

- FAU_GEN.1 – This requirement meets this objective O.AUDIT by ensuring that the TOE maintains a record of defined security related events, including relevant details about the event.
- FAU_GEN.2 – This requirement meets the objective O.AUDIT by ensuring that the TOE associates each auditable event with the identity of the user that caused the event.
- FAU_SAR.1 – This requirement meets the objective O.AUDIT by ensuring that authenticated users are able to review logs.
- FAU_SAR.2 – This requirement meets the objective O.AUDIT by ensuring that only authenticated users are able to read the audit records.
- EXT_WAF_RDR.1 – This requirement meets the objective O.AUDIT by ensuring that only authorized users are able to read the collected events from the web systems.

O.AUTH

- FIA_ATD.1 – This requirement meets the objective O.AUTH by defining attributes of the users that are necessary for a secure identification and authentication mechanism.

- FIA_UAU.2 – This requirement realizes the authentication part of the objective O.AUTH since it requires that each user has to be successfully authenticated.
- FIA_UID.2 – This requirement realizes the identification part of the objective O.AUTH since it requires that each user has to be successfully identified.

O.MANAGE

- FMT_MSA.1 – This requirement meets the objective O.MANAGE by ensuring that only authorized users are able to manage the security attributes.
- FMT_MSA.3 – This requirement meets the objective O.MANAGE by ensuring that no one is allowed to specify alternative initial values to override the default values.
- FMT_MTD.1 – This requirement meets the objective O.MANAGE by ensuring that only authorized users are allowed manage the TSF data.
- FMT_SMF.1 – This requirement meets the objective O.MANAGE by ensuring that the TOE includes administrative functions to facilitate the management of the TSF.
- FMT_SMR.1 – This requirement meets the objective O.MANAGE by ensuring that the TOE associates users with roles to provide access to TSF management functions and data.

O.RESPONSE

- EXT_WAF_RCT.1 – This requirement meets the objective O.RESPONSE by ensuring that the TOE respond to detected attacks against protected web systems by sending an alarm to a configured destination, and taking other actions as specified by the TOE's configuration.

O. ANALYSIS

- EXT_WAF_ANL.1 – This requirement meets the objective O. ANALYSIS by ensuring that the TOE specify the capability to perform analysis functions and to record the results of its analysis.
- EXT_WAF_SDC.1 – This requirement meets the objective O. ANALYSIS by ensuring that the TOE collects and store information about all attacks against protected web systems.

O. PATTERN

- EXT_WAF_ANL.1 – This requirement meets the objective O.PATTERN by ensuring that the TOE specify the capability to perform analysis functions and to record the results of its analysis.
- EXT_WAF_RCT.1 – This requirement meets the objective O.PATTERN by ensuring that the TOE respond to detected attacks against protected web systems by sending an alarm to a configured destination, and taking other actions as specified by the TOE's configuration.
- EXT_WAF_RDR.1 – This requirement meets the objective O.PATTERN by ensuring that only authorized users are able to read the collected events from the web systems.
-

6.3.2 Dependencies of security functional requirements

Table 6.7 – Dependencies of security requirements

Requirement	Dependency	Fulfilled
FAU_GEN.1	FPT_STM.1	FPT_STM.1 is not included because time stamps are provided by the operational environment. An environmental objective states that the TOE will receive reliable timestamps provided by the underlying operating system (OE.OS).
FAU_GEN.2	FIA_UID.1	Yes, by FIA_UID.2 that is hierarchical to FIA_UID.1.
	FAU_GEN.1	Yes
FAU_SAR.1	FAU_GEN.1	Yes
FAU_SAR.2	FAU_SAR.1	Yes
FIA_ATD.1	No dependencies	n/a
FIA_UAU.2	FIA_UID.1	Yes, by FIA_UID.2 that is hierarchical to FIA_UID.1.
FIA_UID.2	No dependencies	n/a
FDP_ACC.1	FDP_ACF.1	Yes
FDP_ACF.1	FDP_ACC.1	Yes
	FMT_MSA.3	Yes
FMT_MSA.1	FDP_ACC.1	Yes
	FMT_SMF.1	Yes
	FMT_SMR.1	Yes
FMT_MSA.3	FMT_MSA.1	Yes
	FMT_SMR.1	Yes
FMT_MTD.1	FMT_SMF.1	Yes
	FMT_SMR.1	Yes
FMT_SMF.1	No dependencies	n/a
FMT_SMR.1	FIA_UID.1	Yes, by FIA_UID.2 that is hierarchical to FIA_UID.1.
EXT_WAF_SDC.1	No dependencies	n/a
EXT_WAF_ANL.1	FPT_STM.1	FPT_STM.1 is not included because time stamps are provided by the operational environment. An environmental objective states that the TOE will receive reliable timestamps provided by the underlying operating system (OE.OS).

Requirement	Dependency	Fulfilled
	EXT_WAF_SDC.1	Yes
EXT_WAF_RCT.1	EXT_WAF_ANL.1	Yes
EXT_WAF_RDR.1	EXT_WAF_SDC.1	Yes
	EXT_WAF_ANL.1	Yes
	FMT_SMR.1	Yes

6.3.3 Rational for the assurance requirements

EAL2 augmented by the component ALC_FLR.2 Basic Flaw Remediation was selected because it is the first time this particular TOE is going to be evaluated. Therefore and in order to keep evaluation efforts reasonable a basic level of independently assured security is required for the TOE.

EAL2 augmented by the component ALC_FLR.2 provides assurance by an analysis of the security functions, using a security-enforcing functional specification, guidance documentation, the basic design of the TOE to understand the security behavior. AVA_VAN.2 provides resistance against attackers with basic attack potential and ensures that the evidence shows that vulnerabilities have been analyzed. The analysis is supported by independent sample testing of the TOE security functions, evidence of developer testing based on the security-enforcing functional specification and basic design, selective independent confirmation of the developer test results, and a vulnerability analysis demonstrating resistance to penetration attackers with a basic attack potential.

7 TOE summary specification

This chapter presents an overview of the security functionality implemented by the TOE.

7.1 SF1 – Security Audit

The Security Audit function provides the TOE with the functionality of generating audit records (FAU_GEN.1, FAU_GEN.2).

The TOE processes and records the following general types of event:

- Internal events—these are divided into audit events and service events. Service events provide information on the status and performance of the TOE. Audit events provide information on TOE User Interface activity, including security-related activity.

The audit log of user actions contains the following information for each entry:

- IP address – IP address of the user who initiated the event,
- User – name of the user who initiated the event,
- View – object of access,
- Action – action performed by the user,
- Item – name of the object,
- ID – event identification number (if applicable),
- Date – date and time when the event occurred.

All actions performed by TOE users that are listed in Table 6.2 are recorded by the Security Audit function. Further all authentication attempts regardless whether those were successful or failed are recorded.

The service event log contains the following information for each entry:

- Level — level of entry state (info, debug, error),
- IP — IP address,
- Node — cluster node name,
- Service — the name of the service for which the event was described,
- Message — description of the service status,
- Timestamp — timestamp of an event.

The Attack log at based mode contains the following information for each entry:

- Date and time of the event,
- Event severity,
- Event name,
- Event description,
- Event tag,
- Client IP-address,
- Client geolocation,
- Request.

Only administrators TOE are allowed to review the PT AF user actions log. For a better finding of special events the user has the possibility to filter the stored events (FAU_SAR.1, FAU_SAR.2).

The Security Audit function works in TOE all the time if TOE is initialized.

7.2 SF2 – Access Control

The TOE provides a Role-based access control policy to control the access of users to objects, based on the membership of this user to groups, the requested operation and the requested object (FDP_ACC.1).

Each identified and authenticated TOE user is a member of a user group which is associated with exactly one role. The user rights depend on the user group, since all members of a user group have the same rights. It is not possible to change the access rights of one specific user but only for a whole user group. Especially a new user can be assigned only to an already existing user group.

When a user attempts to perform an action to an object under the control of the TOE, the TOE decides whether the action is to be permitted based on the following rules (FDP_ACF.1):

- If the requested operation on the requested object is permitted to the group of which the authenticated user is a member, access will be grant,
- else the access will be denied.

Within the certified version of the PT AF the following roles are considered (FMT_SMR.1):

- Administrator – The members of the group that is associated with this role have unrestricted access to all TOE functionalities except of passwords of TOE users. Especially only the members of this group are allowed to create, remove and change users, user groups and object owners.
- Configurator – The members of the group that is associated with this role are responsible to change some network settings and basic settings of policies/server groups.
- Custom - The members of the group that is associated with this role are responsible to manage the TOE functionalities that are allowed to handle.

Table 6.3 provides a detailed overview of the allowed operations on objects by the different roles and therefore user groups.

7.3 SF3 – Security Management

Security management specifies how the TOE allows managing the security functionalities of the TOE. This comprises the following management functions (FMT_SMF.1):

- View the dashboard with all detailed information about attacks and security alerts,
- Scan, exclude and block detected attacks,
- View the information in the autodiscovery wizard tab,
- View, create, edit settings and delete the policies which allow to configure the TOE to monitor a web application
- View, create, edit and delete the web applications which allow to configure

- View, create, edit and delete the services which contain settings that must be specified to protect web applications,
- View, delete and upload the SSL certificates and keys that ensure SSL protection of the websites shielded by the TOE,
- View, create, edit the settings and delete the SSL configurations,
- View, create, edit, copy and delete the rules and the security events which are used in the Rule engine protector,
- View, create, edit and delete the actions which TOE can apply to traffic,
- View, create, edit and delete the tags that are indicators used when generating events and alerts,
- View, switch, clear and train the HMM models using by HMM Protector,
- View, create, edit and delete content security policies using by CSP Protector,
- View, create, edit and delete IP Blacklist,
- View, create, edit and delete the host blacklist,
- View, upload and delete XML Schemas using by XML Protector,
- View, create, edit and delete the form policies to generate a signature,
- View, create, edit and delete the regular expression types,
- View and edit the gateways,
- View, create, edit and delete the upstreams which are the addresses of the servers, protected by the TOE,
- View information about the sniffer configuration,
- View, create, edit and delete the network interface aliases
- View, create, edit and delete the virtual IP addresses,
- View, create, edit and delete the source IP ranges,
- View information about cluster settings,
- View, create, edit and delete the events which are the rules for generating a compound event,
- View, create, edit and delete the alerts which are rules that consist of one or several potential events and allow correlating compound events into a chain,
- View, create, edit and delete the list of ICAP services using for the analysis of the documents downloaded to the protected application
- View, create, edit and delete settings of LDAP services used for logging on to Active Directory,
- View, edit and delete information about suspicious sessions,
- View information about the system events and service statuses on all nodes of a cluster,
- View, create, edit and delete users and user groups,
- View user actions in the TOE,
- View, create, edit and delete active directory settings used to authorize in TOE under domain accounts,
- View the trainer settings,
- View and edit the user interface settings,
- View and edit the user interface security settings,

- View information about the settings of SMTP server,
- View information about the time of rules update and their settings,
- View information about the remote access of TOE status,
- View and create TOE event logs,
- View information about the access log of a VPN connection,
- View information about the log of commands executed by the TOE technical support,
- View information about TOE version and current license,
- View information about the uploaded logs to analysis, upload and delete logs,
- View information about the created reports, download reports, delete reports from the list,
- View information about the report templates, create and delete report templates, edit settings of report templates,
- View information about the report schedule, create and delete the report schedule, edit settings of the report schedule,
- View information about the backups, create and delete backups, edit settings of backups,
- View information, create and delete the backup schedule, edit settings of backup schedule,
- View and use IP Whois tool to get information about an attacker IP-address
- View and use the regular expression tester,
- View and use the trainer debug for testing HM models,
- View and delete information about the used virtual patches.

The management functions cannot be performed by all authenticated users but are assigned to authorized user roles. Those user roles are represented by user groups whereby each authenticated TOE user is a member of exactly one user group.

By default only TOE administrator can set actions available for performing by other TOE users in any tabs of User Interface by choosing available actions for tab to manage security attributes. Table 6.3 provides a detailed overview which management function can be performed by which user role (FMT_MTD.1).

The manage security attributes refers to the ability of the TOE Administrators to modify the available management functions by selecting the security attributes available for each tab for user groups (FMT_MSA.1). Thereby the TOE provides restrictive default values for security attributes that are not allowed to change by any user of the TOE (FMT_MSA.3).

7.4 SF4 – Web Application Firewall

7.4.1 Data Collection

The TOE collects network traffic for the protected web systems. The traffic is used for analysis against configured policies and to detect profile changes for web systems.(EXT_WAF_SDC.1)

7.4.2 Analysis

As network traffic is collected, it is analyzed by the TOE. For network traffic analysis, TOE uses a number of protective mechanisms, which are called protectors. Each type of analysis corresponds to a specific set of protectors (EXT_WAF_ANL.1):

- Signature – HTTP Protector, SQL Injection Protector, Response Filter, Rule Engine, XSS Protector, ICAP Protector,
- Statistical – HMM Protector, DDos Protector, WafJs,
- Proactive – Open Redirect Protector, XML Protector, ACL Protector, LDAP Protector, Blacklist Protector, Session Tracking, JSON Protector, CSP Protector, CSRF Protector.

The protectors respond in this order:

- HTTP Protector,

The protector checks HTTP requests on the low level. Also, the protector analyzes them for RFC compliance.

- HMM Protector,

HMM Protector is a module that uses hidden Markov models for parameter analysis of requests that pass through PT AF. Based on the analyzed data, the system undergoes adaptive training in order to detect attempts to inject nonstandard characters.

- CSRF Protector,

The idea of CSRF is that requests are executed on a vulnerable site on behalf of a victim (for example, editing password or email, adding an administrator). If the victim user visits a malicious web site, a request for some malicious operation (for example, money transfer to the attacker’s account) is sent secretly to the other server (for example, the server of a payment system).

- DDoS Protector,

DDoS Protector reads syslog messages received as UDP datagrams on port 15042. Each datagram is recognized by the protector as a request. The protector creates a single execution thread for each virtual host. A thread involves a monitor, trainer, and detector.

- SQL Injection Protector,

The protector prevents an attack against web systems, which involves cracking a site or DB-based program—specifically, the protector prevents injection of an arbitrary SQL code into a request, which could allow an attacker to read the contents of any tables; delete, modify, or add data; or run arbitrary commands on the victim server.

- XSS Protector,

The protector prevents the attacks of the "Cross-Site Scripting" type. It checks HTTP responses foruser-injected data. Specify the Minimum string length parameter, which sets the minimum length ofa parameter value to check.

- Open Redirect Protector,

The module protects from the attacks of the "Open redirect" type. Open Redirect redirects users to the site, specified in the URI parameters. At the same time, the parameter contents are not validated. This kind of redirection is not safe since it can be used for phishing.

- XML Protector,

The module processes XML documents and SOAP requests.

XML protection is based on the analysis of document contents, before the document is delivered to the server for analysis, and rejection of potentially malicious contents without execution of malicious code on the PT AF level.

- ICAP Protector,

The protector uses ICAP to check incoming and outgoing traffic for malicious content. Specifically, it checks incoming and outgoing files for viruses.

- Rule Engine,

The module checks if parameters match attack signatures to recognize malicious requests.

- CSP Protector,

CSP Protector is a module that provides automated training and enforcement of the content security policy.

- Response Filter,

The protector checks and changes HTTP responses to prevent data leakage and give extra protection using special HTTP headers.

- WafJS,

The protector enables CSRF and XSS prevention on the client side, detects bots (PhantomJS, Selenium) and hacking tools (Acunetix, Burp Suite, ZAP, and so on) by adding a JavaScript script and meta tags to a protected page to detect CSRF.

- ACL Protector,

The ACL Protector extracts authentication data from a request, monitors login attempts, and.manages access of users or user groups to resources. To do this, the protector uses ACL rules (ACL is an access control list), which define the permissions of subjects (a user or user group) to access the application resources. The protector implements an access control model even if the web application does not support access control.

- LDAP Protector

With this protector is possible to get user authentication data and check it in Active Directory when using XML services.

- Blacklist Protector,

This protector provides support of reputation services.

- Session Tracking,

This protector provides support of reputation services. The protector allows you to block users more selectively in comparison with blocking by IP. Also, it identifies Cookie hijacking by multiple criteria and detects Cookie forgery attempts.

- JSON Protector,

JSON protection is based on the analysis of document content, before the document is delivered to the web server for analysis, and exclusion of potentially malicious content.

7.4.3 Reaction

TOE, regardless of the mode of operation, fixes in Attack Log all attacks directed at the protected applications.

The response to the detected attack depends on the protector settings that detected this type of attack.

When an attack is detected, the associated actions are performed as follows (EXT_WAF_RCT.1):

- Log to DB – The Log To DB action logs a detected attack to the Elasticsearch database,
- Sanitize payload – When the Sanitize payload action is triggered, the input data detected in a response will be escaped or replaced with a predefined string,
- Safe redirect – The Safe redirect action is intended to safely redirect the client to the predefined page,
- Send to ArcSight, Send to syslog, Send to QRadar – With these actions user can send alert information to an external log system,
- Block IP – If the check is triggered, the attacker's IP address will be blocked,
- Send to correlator – The Send to Correlator action sends information about events to the PT AF system correlator without sending it to Elasticsearch,
- Send to Blacklist – The action Send to Blacklist is used for blocking traffic by IP address on the application level,
- Block request – With the Block request action user can specify a template of the page, which will be displayed when the system blocks requests.

7.4.4 Restricted Data Review

The TOE provides capability for Administrators to view security events via Dashboard (EXT_WAF_RDR.1).

The dashboard can run in two modes—basic and advanced. In the basic mode, is possible to configure only filters, while the advanced mode allows a wide range of actions to configure data representation and output.

Each HTTP request displayed on the PT AF Dashboard is assigned one or several tags that describe the event. The HTTP requests are displayed in the Attacks panel of the Dashboard. These HTTP requests are called events.

TOE can generate reports of the following formats: CSV (a ZIP archive), PDF, DOC, or HTML.

The contents of the section can include in a report are shown in the table below.

Table 7.1 – Sections of a generated report

Section	Section contents	Description
General	Title	Report name, analyzed period
	Annotations	Introductory part of the report
	System description	Description of the PT AF system
	Attack analysis	Attack description

Section	Section contents	Description
	About Positive Technologies	Information about JSC Positive Technologies
Attacks	Distribution by time	Chart and table showing attack distribution by time
	Distribution by tag	Chart showing attack distribution by its type (tag)
	Distribution of N tag attacks by time	The system generates a chart and table, both sorted by date for each attack type specified. If types are not specified, the system creates a chart and table for all types available in the time interval analyzed
	Distribution by attacking IP address	Chart and table showing attack distribution by source IP address
	Distribution by target IP address and port	Chart and table showing attack distribution by target IP address and port
	Attack count by profile, severity, and tag	For each policy, the system generates a summary table that contains the number of threats of each severity level sorted by types (attacks, vulnerabilities,tools)
	Differential analysis	The system generates a table that contains the number of attacks that happened during the current time interval, and the change rate compared to the previous time interval of the same duration for each pair "attack type–target parameter"
	Targeted sections of web applications	The table is generated for each date and contains the pairs "attack type–path and number of attacks"
	Targeted parameters	The table is generated for each date and contains attack number for each target parameter
	Top 10 source countries	Tables showing attack distribution by source country
	Top 10 user agents	Tables showing attack distribution by user agent
	Top 10 source IP addresses including tags	For each source IP address the system generates a table that contains the number and period of attacks sorted by types
Alerts	Table	Alert list
	OWASP attack classification	Appendix. Attack dictionary

All data is presented in such a manner that it can be read and the contents of the data can be interpreted; thus the reader can understand the content of the information presented.

7.5 SF5 – Identification and Authentication

This security functionality requires each user to be successfully identified and authenticated before allowing any other actions on behalf of that user (FIA_UAU.2, FIA_UID.2).

Within the creation of new users it is enforced that the authentication mechanism, the login credentials and the membership of a group is determined for that user (FIA_ATD.1).

When a user attempts to communicate with the TOE he has first of all to login to the TOE.

The PT AF authentication mechanism is maintained by the TOE itself. For every PT AF Authentication login the TOE maintains a login name and a password. The password is not stored in plain text, but hashed using the hash function performed by the underlying OS according to PBKDF2 standard (FIA_ATD.1).

Each PT AF Authentication login name and the corresponding hash of the password are stored in a table within the connected database.

7.6 Rationale on TOE specification

The specification of the TOE security functions refers directly to the TOE security requirements. The following table displays the correlation between security requirements and security functions.

Table 7.2 – Security Requirements vs. Security Functions

Security Requirements vs. Security Functions	**Security Audit**	**Access Control**	**Security Management**	**Web Application Firewall**	**Identification and Authentication**
FAU_GEN.1	X				
FAU_GEN.2	X				
FAU_SAR.1	X				
FAU_SAR.2	X				
FIA_ATD.1					X
FIA_UAU.2					X
FIA_UID.2					X
FDP_ACC.1		X			
FDP_ACF.1		X			
FMT_MSA.1			X		
FMT_MSA.3		X			
FMT_MTD.1			X		
FMT_SMF.1			X		
FMT_SMR.1		X			
EXT_WAF_SDC.1				X	

Security Requirements vs. Security Functions	**Security Audit**	**Access Control**	**Security Management**	**Web Application Firewall**	**Identification and Authentication**
EXT_WAF_ANL.1				X	
EXT_WAF_RCT.1				X	
EXT_WAF_RDR.1				X	

8 Appendix

8.1 References

[CC] *Common Criteria for Information Technology Security Evaluation*, version 3.1, revision 4
Part 1: Introduction and general model, CCMB-2012-09-001*,*
Part 2: Security functional requirements, CCMB-2012-09-002*,*
Part 3: Security Assurance Requirements, CCMB-2012-09-003.

8.2 Acronyms

CC	Common Criteria
DB	Database
EAL	Evaluation Assurance Level
IT	Information Technology
n/a	not applicable
OS	Operating System
OSP	Organizational Security Policy
PT AF	PT Application Firewall
SAR	Security Assurance Requirement
SF	Security Function
SFP	Security Function Policy
SFR	Security Functional Requirement
ST	Security Target
TOE	Target of Evaluation
TSF	TOE Security Functions
TSFI	TSF Interface