Microsoft®
Forefront™
Unified Access Gateway

Microsoft Forefront UAG 2010 Common Criteria Evaluation

Security Target

Microsoft Forefront Unified Access Gateway Team

Author:	Microsoft Corp.
Version:	1.0
Last Saved:	2011-03-10
File Name:	MS_UAG_ST_1.0.docx

Abstract

This document is the ST (Security Target) of Forefront UAG 2010 SP1 Common Criteria Certification.

Keywords

CC, ST, Common Criteria, Gateway, Security Target

Revision History

Date	Version	Author	Edit
10-Mar-2011	1.0	Microsoft Corp.	Final version

This page intentionally left blank

Table of Contents

List of Tables

List of Figures

Page

1 Introduction

This chapter contains document management and overview information. The Security Target (ST) identification provides the labeling and descriptive information necessary to identify, catalogue, register, and cross-reference a ST. The TOE overview summarizes the ST in narrative form and provides information for a potential user to determine whether Microsoft Forefront Unified Access Gateway (UAG) is of interest. The overview can also be used as a standalone abstract for ST catalogues and registers.

UAG is basically a VPN gateway which is used to build up a virtual private network as depicted below. A Virtual Private Network, or VPN, is a private communication network communicating over a public network, i.e. the Internet. Normally, a local network is protected against unauthorized access from the public network by means of a firewall which limits the permitted types of traffic. The TOE provides remote authorized users up to a full connection into the local network without bypassing this protection against unauthorized users.

UAG provides comprehensive, secure remote access to corporate resources for employees, partners, and vendors on both managed and unmanaged PCs and mobile devices. Utilizing a combination of connectivity options, ranging from SSL VPN to DirectAccess, as well as built in configurations and policies, UAG provides centralized management of an organization’s complete anywhere access offering.

1.1 ST Reference

ST Title:	*Microsoft Forefront* UAG 2010 Common Criteria Evaluation - Security Target
ST Version:	1.0
ST Date:	2011-03-10

1.2 TOE Reference

TOE Identification:	*Microsoft Forefront Unified Access Gateway 2010 (CC)* and its related guidance documentation
TOE Version:	4.0.1752.10000 (English language)
Cert. ID:	BSI-DSZ-CC-0678
TOE Platform:	Windows Server 2008 R2 (English) – 64bit[1]
CC Identification:	Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 3 as of July 2009 for parts I, II and III. ([CC])

[1] Whenever “Windows Server” is referenced in this document it refers to this version only unless explicitly stated.

Evaluation Assurance Level:	EAL2 augmented by ALC_FLR.3
PP Conformance:	none

1.3 TOE Overview

This chapter presents a general overview of Microsoft Forefront Unified Access Gateway (UAG)[2].

UAG provides remote access to applications, networks, and internal resources from diverse client endpoints through a single point of entry. The product type is an application layer gateway.

Forefront Unified Access Gateway (UAG) delivers comprehensive, secure remote access to corporate resources for employees, partners, and vendors on both managed and unmanaged PCs and mobile devices. Utilizing a combination of connectivity options, ranging from SSL VPN to DirectAccess, as well as built in configurations and policies, UAG provides centralized management of an organization’s complete anywhere access offering.

Features of UAG are:

- Anywhere Access: Forefront Unified Access Gateway makes it easier for organizations to deliver secure remote access to their applications and resources and improve employee and partner productivity, by combining an intelligent access policy engine and consolidating a variety of connectivity options including SSL VPN and Direct Access.
- Integrated Security: Forefront Unified Access Gateway improves the security in remote access scenarios by enforcing granular access controls and policies that are tailored to the applications being published, the identity of the user and the health status of the device being used. UAG further improves security by enabling strong authentication to applications and mitigating the risks of downloaded data from unmanaged devices.
- Simplified Management: With Forefront Unified Access Gateway, administrators have a single platform through which to deliver and manage remote access. With built in policies and configurations for common applications and devices, administrators gain more control, more efficient management, greater visibility, and lower total cost of ownership.

The evaluated TOE comprises identification and authentication delegation for users, enforcement of access control to published web applications, establishment of a secure channel over HTTPS, management, audit generation and audit review.

1.4 TOE Description

The TOE, a secure gateway server, is the main part of UAG (the logical scope and boundary are described in chapter 1.4.2) that helps to provide secure connectivity. It is an integrated

[2] short: „UAG“

solution for virtual private networking. UAG can be installed as a dedicated gateway that runs on Windows Server 2008 (English) 64bit operating system. The TOE provides remote authorized users up to a full connection into the local network without bypassing this protection against unauthorized users. This connection is established through a so called tunnel between a gateway on the side of the network and a client on the side of the remote user, which is a reduced form of a gateway. The TOE provides the following functionality:

- Identifying and authenticating remote users,
- Making internal web applications and resources available to remote endpoints by publishing them in an application Web site or portal,
- Building up tunnels between the TOE and the client using agreed cryptographic algorithms.

The traffic is carried on public networking infrastructure using standard protocols. Cryptographic protocols (SSL/TLS) provide the necessary confidentiality (preventing snooping), sender authentication (preventing identity spoofing), and message integrity (preventing message alteration) to achieve the privacy intended. Additionally the TOE creates an audit trail and provides a management console.

Windows Server stores the identification and authentication data for all known administrators and maintains a method of associating human users with the authorized administrator role. The TOE itself offers no additional identification and authentication methods for administrators.

The next chapters describe the physical scope and boundary and the functionalities of the TOE.

1.4.1 Physical scope and boundary

The TOE is delivered in a package which consists of:

- The software package “Microsoft Forefront Unified Access Gateway 2010” delivered as ISO image,
- A manual (a Windows Help File), which is delivered as part of the software package and installed on the host system within the TOE [MSUAG],
- A Guidance Addendum [MSUAG_ADD] delivered via the UAG 2010 Common Criteria product page (see [WEBUAG]).

The website [WEBUAG] contains additional information about the TOE and its evaluated configuration. Also the guidance addendum that describes the specific aspects of the certified version can be obtained via this website. The guidance addendum extends the general guidance of UAG 2010 that ships along with the product in form of a help file. This website shall be visited before using the TOE.

The TOE configuration is a single machine, which comprises the evaluated TOE and non-evaluated components. Microsoft Forefront Threat Management Gateway (TMG) is part of the non-evaluated components. TMG is part of the UAG DVD and will be installed automatically during the installation of the TOE.

The TOE is running on

- Windows Server 2008 R2 (English) - 64bit.

which has been used as underlying operating system for evaluation.

The TOE relies on functionality of the Windows Server Operating System and TMG and has the following hardware requirements:

Table 1.1 – Hardware Requirements

Aspect	Requirement
CPU	64bit, 2.66 GHz or faster, dual core
RAM	4 GB, 800 MHz
Hard Disk	Approx 2500 MB of free space not including the OS, NTFS formatted
Other	Two or more network adapters

The evaluated functionality respectively the TOE (the logical scope) is stated in the following chapter 1.4.2. In particular Figure 1.1 shows the demarcation of the TOE respectively UAG.

1.4.2 Logical scope and boundary

The logical scope and boundary of the TOE is subdivided into the following major functions of the TOE:

- Access Control
- Information Protection
- Audit
- Management

1.4.2.1 Access Control

Identification and authentication is one of the main security functionalities of a gateway. The TOE provides remote authorized users an https connection into the local network without bypassing this protection against unauthorized users. Thus the TOE has to identify and authenticate remote users on behalf of a security attribute before gaining access to the network.

The TOE provides access control to web applications based on Access Control Lists (ACL).

The TOE supports various Identification and Authentication methods. Chapter 7.1 gives an overview about supported and evaluated authentication methods.

1.4.2.2 Information Protection

The TOE protects information of transmitted data by enforcing SSL/TLS secured communication (HTTPS).

1.4.2.3 Audit

The TOE details Forefront UAG events and errors logs in a built-in reporter and local XML based logging formats. Stored audit records can be reviewed.

It further provides logging to the Windows Event Log.

1.4.2.4 Management

UAG provides management of the TOE through the Forefront UAG Management. The GUI exposes all administration functionality necessary for the administering the TOE.

1.4.3 TOE Demarcation Summary

For better understanding the boundaries of the TOE are summarized in Figure 1.1. It shows the TOE with its security functionalities:

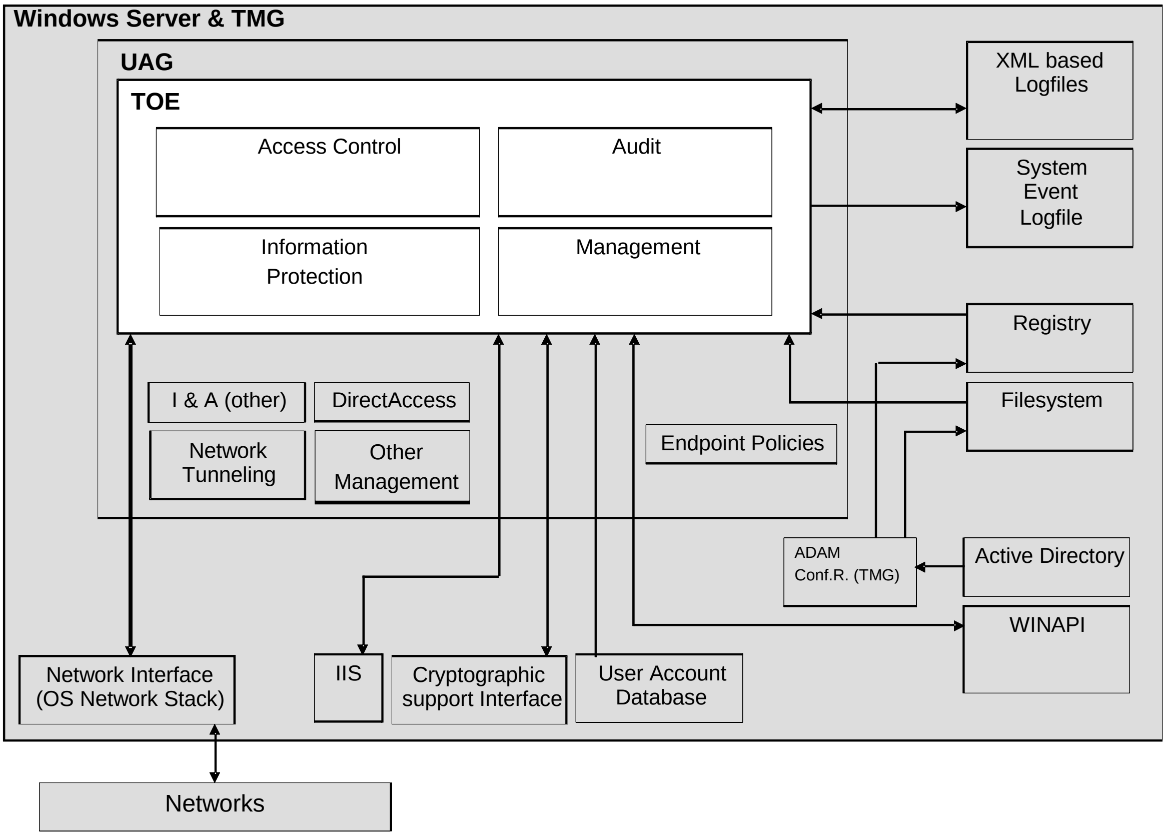
- Access Control
- Information Protection
- Audit
- Management

The additional features of UAG which are not part of the evaluation:non-evaluated Identification & Authentication functionality, Network tunneling, Direct Access, endpoint policies, other Management (like Wizards and non evaluated features) and the used functionalities of the underlying Windows Server operating system. The arrows show the interfaces between the TOE and the operating system, the arrowheads show the direction of possible information flow. The TOE uses the file system to store XML based log files and the Windows event log file, which is protected for unauthorized access by the file system. The configuration is read from files in the file system which have been converted from data read from the registry and files system, which has been replicated from Active Directory to the registry and file system using the ADAM Configuration Receiver (provided by TMG). The user account database provides the information required by the I&A functionality of the TOE which is also read via the network interface from an external authentication server like an Active Directory or LDAP server. The cryptographic support interface supports the required cryptographic algorithms. The network interface is needed for transmitting data to the different networks. The Windows API (WinAPI) provides low level functions which are used by the TOE.

The TOE further uses functionality of the Internet Information Server (IIS, part of Windows) to expose web content and published applications to the user clients.

For information purpose and better understanding the interfaces between ADAM Configuration Receiver and Active Directory, Filesystem and Registry are also shown in the picture below.

Figure 1.1 – TOE demarcation



ADAMConf.R. = ADAM Configuration Receiver (provided by TMG)

1.5 Conventions

For this Security Target the following conventions are used:

The CC allows several operations to be performed on functional requirements; *refinement*, *selection*, *assignment*, and *iteration* are defined in part 1 of [CC]. The following conventions are used in this ST (the performed operation is always written in italics):

The **refinement** operation is used to add detail to a requirement, and thus further restricts a requirement. Refinement of security requirements is denoted by prefix “refinement”.

The **selection** operation is used to select one or more options provided by the CC in stating a requirement. Selections that have been made are denoted by prefix “selection”.

The **assignment** operation is used to assign a specific value to an unspecified parameter, such as the length of a password. Assignments that have been made are denoted by prefix “assignment”.

The **iteration** operation is used when a component is repeated with varying operations. Iterations are indicated by the use of parentheses “()” in the component identification and by parentheses “()” and an abbreviation in the component name.

The CC paradigm also allows protection profile and security target authors to create their own requirements. Such requirements are termed ‘explicit requirements’ and are permitted if the CC does not offer suitable requirements to meet the authors’ needs.

Explicit requirements must be identified and are required to use the CC class/family/component model in articulating the requirements. In this ST, explicit requirements will be indicated with the prefix “EXT_” followed by the component name.

2 Conformance Claims

This chapter contains the following sections:

- CC conformance claims
- PP claim
- Package claim

2.1 CC Conformance Claims

This Security Target claims to be conformant to the Common Criteria 3.1 [CC]:

- Part 2 extended to Common Criteria 3.1 Revision 3, released July 2009

  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 Common Criteria 3.1 Revision 3, released July 2009

  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 Package Claim

This ST claims conformance to the assurance requirements package EAL2 augmented by ALC_FLR.3. This ST does not claim conformance to any other packages.

2.3 PP Claim

This ST does not claim conformance to any PP.

3 Security Problem Definition

This chapter aims to clarify the security problems that UAG is intended to solve, by describing any assumptions about the security aspects of the environment and/or of the manner in which the TOE is intended to be used, organizational security policies (OSPs) that have to be enforced, and any known or assumed threats to the assets against which protection within the TOE or its environment is required.

3.1 Definition of terms

This section is added to define the terms like subjects, objects and operations, that are used in the Security Objectives of the Operational Environment and SFRs.

3.1.1 Subjects

S.USER: User of the TOE, typically a remote user or an administrator of a remote network.

S.ADMIN: Administrator, authorized to modify the list of authorized users and administrating the corresponding security attributes of S.USER.

3.1.2 Objects

O.NETWORK local network to which the TOE provides the connection.

3.1.3 Operations

R.CONNECT establishes connection between S.USER and the O.NETWORK by a so-called tunnel. SSL/TLS is used to protect the tunnel.

3.1.4 Security Function Policies

SFP.ACCESS defines the access control policy for authenticated users that want to access resources published by the TOE.

3.2 Assets

The TOE protects the following types of data concerning confidentiality and integrity:

- Data transmitted between internal IT entities (published applications) and endpoints (user clients)
- Audit records
- Internal IT entities itself

The assets under attack are: internal IT entities which are protected by the TOE. In general, the threat agent (attacker) includes, but is not limited to:

- Not authorized persons

3.3 Threats

Threats to the TOE are defined in Table 3.1 below.

Table 3.1 – Threats

#	Threat	Description
1	T.AUDFUL	An attacker may cause audit records to be lost or prevent future records from being recorded by taking actions to exhaust audit storage capacity, thus masking attackers' actions. This might be a result of a strange denial of service attack.
2	T.MEDIAT	An attacker may send impermissible information through the TOE, which results in the exploitation of resources on the internal network and/or gathering of information a person is not authorized for.
3	T.NOAUTH	An attacker may attempt to bypass the security of the TOE so as to access and use security functionality and/or non-security functionality provided by the TOE.

3.4 Organizational Security Policies

Security policies to be fulfilled by the TOE are defined in Table 3.2 below.

Table 3.2 – Security Policies addressed by the TOE

#	Policy Name	Description
1	P.AUDACC	S.ADMIN must be accountable for the actions that they conduct. Therefore audit records must contain sufficient information to prevent an attacker to escape detection.

3.5 Assumptions

Table 3.3 lists the TOE Secure Usage Assumptions for the IT and non-IT environment and intended usage.

Table 3.3 – Assumptions for the IT and non-IT Environment and intended usage

#	Assumption Name	Description
1	A.ADMIN	The security attributes of S.USER are adequately imported and handled by S.ADMIN.
2	A.CLIENT	The remote user S.USER utilizes an IT product (client or another gateway) for the connection to the TOE.
3	A.DIRECT	The TOE is available to S.ADMIN only. Personnel who has physical access to the TOE and can log in the operating system is assumed to act as an authorized TOE administrator.
4	A.ENV	The environment implements following functionality:

#	Assumption Name	Description
		local identification and authentication of user credentials used I&A, reliable time stamp (log file audit), file protection (for log file access protection, registry protection, and AD protection), cryptographic support, administration access control, reliable AD implementation, secure connection to the TOE (e.g. using SSL/TLS), reliable packet/application filtering mechanism (using the underlying TMG)
5	A.GENPUR	The TOE stores and executes security-relevant applications only. It stores only data required for its secure operation. Nevertheless the underlying operating system may provide additional applications required for administrating the TOE or the operating system.
6	A.NOEVIL	S.ADMIN is non-hostile and follows all administrator guidance.
7	A.PHYSEC	The TOE is physically secure. Only authorized personal has physical access to the system which hosts the TOE.
8	A.PRIV_DATA	The private data belonging to the security attributes of S.USER are generated and handled in a secure manner.
9	A.SINGEN	Information should not flow among O.NETWORK and public networks unless it passes through the TOE. Thereby S.ADMIN has to guarantee an adequate integration of the TOE into the environment.

4 Security Objectives

This chapter contains the following sections:

- Security objectives for the TOE
- Security objectives for the operational environment
- Security objectives rationale

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
1	O.ACCOUN	The TOE must provide S.USER accountability for information flows through the TOE.
2	O.AUDREC	The TOE must provide a means to record a readable audit trail of security-related events, with accurate dates and times, and a means to search and sort the audit trail. The TOE ensures that no records are dropped because of low storage capacity.
3	O.IDAUTH	The TOE must uniquely identify and authenticate the claimed identity of S.USER, before granting a user access to TOE functions. The TOE has to request user credentials from S.CLIENT and has to call a function in the operating system to verify these.
4	O.MEDIAT	The TOE must mediate the flow of all information from S.USER on a connected network to users on another connected network.
5	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.

4.2 Security objectives for the operational environment

Table 4.2 lists security objectives for the Environment (covers objectives for the IT-Environment and non IT-Environment).

Table 4.2 – Security Objectives for the Environment

#	Objective Name	Objective Description
1	OE.ADMIN	The security attributes of S.USER should be adequately imported and handled by S.ADMIN.
2	OE.CLIENT	The remote user S.USER should utilize an IT product (client or another gateway) for the connection to the TOE.
3	OE.DIRECT	The TOE should be available to S.ADMIN only.

#	Objective Name	Objective Description
4	OE.ENV	The environment should implement following functionality: local identification and authentication of user credentials used for I&A, reliable time stamp (log file audit), file protection (for log file access protection, registry protection, and AD protection), cryptographic support, administration access control, reliable AD implementation, secure connection to the TOE (e.g. using SSL/TLS), reliable packet/application filtering mechanism (using the underlying TMG)
5	OE.GENPUR	The environment should store and execute security-relevant applications only and should store only data required for its secure operation.
6	OE.NOEVIL	S.ADMIN should be non-hostile and should follow all administrator guidance.
7	OE.PHYSEC	The system which hosts the TOE should be physically secure.
8	OE.PRIV_DATA	The private data belonging to the security attributes of S.USER are generated and handled in a secure manner.
9	OE.SINGEN	Information should not flow among O.NETWORK and public networks unless it passes through the TOE. Thereby S.ADMIN has to guarantee an adequate integration of the TOE into the environment.

4.3 Security objectives rationale

This table maps assumptions, OSPs and threats to objectives, demonstrating that all assumptions and threats are mapped to at least one objective and vice versa. A discussion of the rationale for threat mappings is provided below.

Table 4.3 – Security Objectives rationale

Threats and Assumptions vs. Security Objectives	O.ACCOUN	O.AUDREC	O.IDAUTH	O.MEDIAT	O.SECSTA	OE.ADMIN	OE.CLIENT	OE.DIRECT	OE.ENV	OE.GENPUR	OE.NOEVIL	OE.PHYSEC	OE.PRIV_DATA	OE.SINGEN
T.AUDFUL		X												
T.MEDIAT				X										
T.NOAUTH			X		X									
P.AUDACC	X	X												
A.ADMIN						X								
A.CLIENT							X							
A.DIRECT								X						
A.ENV									X					
A.GENPUR										X				
A.NOEVIL											X			
A.PHYSEC												X		
A.PRIV_DATA													X	
A.SINGEN														X

Note:

The security objectives for the environment are a restatement of the assumptions for the environment.

T.AUDFUL: “An attacker may cause audit records to be lost or prevent future records from being recorded by taking actions to exhaust audit storage capacity, thus masking attackers’ actions.”

T.AUDFUL is countered by O.AUDREC because the security objective ensures that the TOE records a reliable readable audit trail and that no records are left because of less storage capacity.

T.MEDIAT: “An attacker may send impermissible information through the TOE, which results in the exploitation of resources on the internal network and/or gathering of information he is not authorized for.”

T.MEDIAT is countered by O.MEDIAT because the security objective ensures that the TOE mediates the flow of all information from users on the connected network to users on another connected network.

T.NOAUTH: “An attacker may attempt to bypass the security of the TOE so as to access and use security functions and/or non-security functions provided by the TOE.”

T.NOAUTH is countered by O.IDAUTH, O.SECSTA because the security objective ensures that the user has to authenticate before access is granted to TOE functions and the TOE ensures that it does not compromise its resources or those of any connected network.

P.AUDACC: “S.ADMIN must be accountable for the actions that they conduct. Therefore audit records must contain sufficient information to prevent an attacker to escape detection."

P.AUDACC is countered by O.AUDREC, O.ACCOUN because the security objective ensures that a person is identified to make the person accountable for the action and that this action is logged in the audit trail.

O.ACCOUN: This security objective is necessary to counter the policy: P.AUDACC because it requires that users are accountable for information flows through the TOE and that authorized administrators are accountable for the use of security functions related to audit.

O.AUDREC: This security objective is necessary to counter the policy: P.AUDACC by requiring a readable audit trail and a means to search and sort the information contained in the audit trail and T.AUDFUL by requiring that no records are left because of not enough storage capacity.

O.IDAUTH: This security objective is necessary to counter the threat T.NOAUTH. It requires that users be uniquely identified before accessing the TOE and sending information through the TOE.

O.MEDIAT: This security objective is necessary to counter the threats: 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: 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 and thus counters the threats: T.NOAUTH.

5 Extended Components Definition

This chapter defines TOE security functional requirements which are not part of CC 3.1 part 2. There are no extended security assurance requirements defined in this ST.

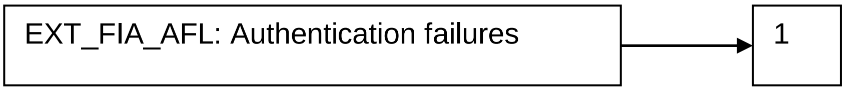
5.1 Definition of functional family EXT_FIA_AFL

5.1.1 EXT_FIA_AFL Authentication failures

Family Behavior:

This family contains requirements for defining values to specify the number of unsuccessful authentication attempts and the TSF actions in cases of authentication attempt failures. Parameters include, but are not limited to, the number of failed authentication attempts and time thresholds.

Component leveling:



EXT_FIA_AFL.1 Authentication failure handling, requires that the TSF be able to terminate the session establishment process after a specified number of unsuccessful user authentication attempts. It also requires that, after termination of the session establishment process, the TSF be able to disable the user account or the point of entry (e.g. workstation) from which the attempts were made until an administrator-defined condition occurs.

Management: EXT_FIA_AFL.1

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

a) management of the threshold for unsuccessful authentication attempts;

b) management of actions to be taken in the event of an authentication failure.

Audit: EXT_FIA_AFL.1

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

a) Minimal: the reaching of the threshold for the unsuccessful authentication attempts and the actions (e.g. disabling of a terminal) taken and the subsequent, if appropriate, restoration to the normal state (e.g. re-enabling of a terminal).

EXT_FIA_AFL.1 Authentication failure handling

Hierarchical to: No other components.

EXT_FIA_AFL.1.1 The TSF shall detect when [*selection: [assignment: positive integer number*], *an administrator configurable positive integer within* [*assignment*: *range of acceptable values*]] unsuccessful authentication attempts occur related to [*assignment: list of authentication events*].

EXT_FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been [*selection: met or surpassed*], the TSF shall [*assignment: list of actions*].

EXT_FIA_AFL.1.3 The TOE shall handle the authentication failure after the verification has failed.

Dependencies: EXT_FIA_UAU.1 Timing of authentication

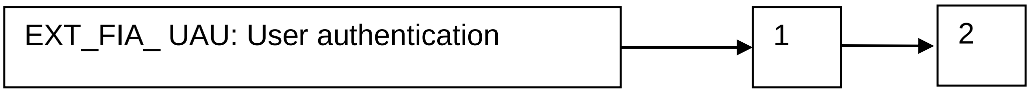
5.2 Definition of functional family EXT_FIA_UAU

5.2.1 EXT_FIA_UAU User authentication

Family Behavior:

This family defines the types of user authentication mechanisms supported by the TSF. This family also defines the required attributes on which the user authentication mechanisms must be based.

Component leveling:



EXT_FIA_UAU.1 Timing of authentication, allows a user to perform certain actions prior to the authentication of the user's identity.

EXT_FIA_UAU.2 User authentication before any action, requires that users are authenticated before any action will be allowed by the TSF.

Management: EXT_FIA_UAU.1

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

a) management of the authentication data by an administrator;

b) management of the authentication data by the associated user;

c) managing the list of actions that can be taken before the user is authenticated.

Management: EXT_FIA_UAU.2

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

a) management of the authentication data by an administrator;

b) management of the authentication data by the user associated with this data.

Audit: EXT_FIA_UAU.1

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

a) Minimal: Unsuccessful use of the authentication mechanism;

b) Basic: All use of the authentication mechanism;

c) Detailed: All TSF mediated actions performed before authentication of the user.

Audit: EXT_FIA_UAU.2

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

a) Minimal: Unsuccessful use of the authentication mechanism;

b) Basic: All use of the authentication mechanism.

EXT_FIA_UAU.1 Timing of authentication

Hierarchical to: No other components.

EXT_FIA_UAU.1.1 The TSF shall allow [*assignment: list of TSF mediated actions*] on behalf of the user to be performed before the user is authenticated.

EXT_FIA_UAU.1.2 The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

Dependencies: EXT_FIA_UID.1 Timing of identification

EXT_FIA_UAU.2 User authentication before any action

Hierarchical to: EXT_FIA_UAU.1 Timing of authentication

EXT_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.

EXT_FIA_UAU.2.2 The TOE shall initiate the verification of [*assignment: list of data*].

Dependencies: EXT_FIA_UID.1 Timing of identification

5.3 Definition of functional family EXT_FIA_UID

5.3.1 EXT_FIA_UID User identification

Family Behavior:

This family defines the conditions under which users shall be required to identify themselves before performing any other actions that are to be mediated by the TSF and which require user identification.

Component leveling:

EXT_FIA_ UID: User Identifikation
1
2

EXT_FIA_UID.1 Timing of identification, allows users to perform certain actions before being identified by the TSF.

EXT_FIA_UID.2 User identification before any action, requires that users identify themselves before any action will be allowed by the TSF.

Management: EXT_FIA_UID.1

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

a) the management of the user identities;

b) if an authorized administrator can change the actions allowed before identification, the managing of the action lists.

Management: EXT_FIA_UID.2

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

a) the management of the user identities.

Audit: EXT_FIA_UID.1, EXT_FIA_UID.2

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

a) Minimal: Unsuccessful use of the user identification mechanism, including the user identity provided;

b) Basic: All use of the user identification mechanism, including the user identity provided.

EXT_FIA_UID.1 Timing of identification

Hierarchical to: No other components.

EXT_FIA_UID.1.1 The TSF shall allow [*assignment: list of TSF-mediated actions*] on behalf of the user to be performed before the user is identified.

EXT_FIA_UID.1.2 The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

Dependencies: No dependencies.

EXT_FIA_UID.2 User identification before any action

Hierarchical to: EXT_FIA_UID.1 Timing of identification

EXT_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.

Dependencies: No dependencies.

6 Security Requirements

6.1 Security Functional Requirements

This chapter defines the TOE security functional requirements. A list of the requirements is provided in Table 6.1. The full text of the security functional requirements is contained below.

Table 6.1 – TOE Security Functional Requirements

#	Functional Requirement	Title	Dependencies
Access Control			
1	EXT_FIA_AFL.1	Authentication failure handling	EXT_FIA_UAU.1
2	EXT_FIA_UAU.2	User authentication before any action	EXT_FIA_UID.1
3	EXT_FIA_UID.2	User identification before any action	none
4	FDP_ACC.1	Subset access control	ADP_ACF.1
5	FDP_ACF.1	Security attribute based access control	FDP_ACC.1 FMT_MSA.3
Information Protection			
6	FTP_ITC.1	Inter-TSF trusted channel	none
7	FMT_MSA.3	Static attribute initialization	FMT_MSA.1 FMT_SMR.1
Audit			
8	FAU_GEN.1	Audit data generation	FPT_STM.1
9	FAU_SAR.1	Audit review	FAU_GEN.1
10	FAU_SAR.3	Selectable audit review	FAU_SAR.1
11	FAU_STG.3	Action in case of possible audit data loss	FAU_STG.1
Management			
12	FMT_MSA.1	Management of secure attributes	FDP_ACC.1 FMT_SMR.1 FMT_SMF.1
13	FMT_SMF.1	Specification of Management Functions	none

Note:
FPT_STM.1, FAU_STG.1, and FMT_SMR.1 are considered in the IT environment.

6.1.1 Class FAU – Security audit

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 audit functions;

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

c) [*assignment: the events specified 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, 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 ST, [*assignment: information specified in column four of* Table 6.2].

Table 6.2 – Auditable Events

Functional Component	**Level**	**Auditable Event**	**Additional Audit Record Contents**
EXT_FIA_AFL.1	minimal	The reaching of the threshold for unsuccessful authentication attempts.	The user identities provided to the TOE
EXT_FIA_UAU.2	basic	All use of the user authentication mechanism.	The user identities provided to the TOE
EXT_FIA_UID.2	basic	All use of the user identification mechanism.	The user identities provided to the TOE
FAU_GEN.1	(none)	-	-
FAU_SAR.1	minimal	No auditable events	-
FAU_SAR.3	minimal	No auditable events	-
FAU_STG.3	minimal	No auditable events	-
FDP_ACC.1	(none)	-	-
FDP_ACF.1	basic	All requests to perform an operation on an object covered by the SFP.	Session Id, source IP address and trunk (target).
FMT_MSA.1	minimal	No auditable events	-
FMT_MSA.3	minimal	No auditable events	-
FMT_SMF.1	minimal	Use of management functions	The fact that the configuration was changed
FTP_ITC.1	basic	All attempted uses of the trusted channel functions. Identification of the initiator and target of all trusted channel functions.	Session ID, source IP address and trunk (target).

Application Notes:

The auditable event FMT_SMR.1 “Minimal: modifications to the group of users that are part of a role” is not part of the TOE (the functional component FMT_SMR.1 is part of the environment). User accounts are managed by the underlying operating system.

The timestamp is provided by the underlying operating system and used for logging. FPT_STM.1 is part of the environment.

FAU_SAR.1 Audit review

FAU_SAR.1.1 The TSF shall provide [*assignment: S.ADMIN*] with the capability to read [*assignment: 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.

Application note:

The TOE preprocesses the audit data in order to allow the Web Monitor to display the items.

FAU_SAR.3 Selectable audit review

FAU_SAR.3.1 The TSF shall provide the ability to perform [*selection: filtering, searches, sorting*] of audit data based on:

[*assignment:*

a) time and date;

b) *trunk;*

c) *category;*

d) *severity*

e) *type*].

<u>Application note:</u>

The TOE preprocesses the audit data in order to allow the Web Monitor to display the filtered, selected or ordered items.

FAU_STG.3 Action in case of possible audit data loss

FAU_STG.3.1 The TSF shall take [*assignment: overwrite the oldest stored audit records*] if the audit trail exceeds [*assignment: a defined capacity limit*].

6.1.2 Class FIA – Identification and authentication

EXT_FIA_AFL.1 Authentication failure handling

EXT_FIA_AFL.1.1 The TSF shall detect when [*selection*: [*assignment: one or more*]] unsuccessful authentication attempts occur related to [*assignment: Forms-based authentication].*

EXT_FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been [*selection: met*], the TSF shall [*assignment: create a log file entry*].

EXT_FIA_AFL.1.3 The TOE shall handle the authentication failure after the verification has failed.

<u>Note:</u>

Forms-based authentication is used in the in the Front-End Authentication process (see chapter 7.1 for more information).

Unlike FIA_AFL.1 (component from CC part II) the required verification of the user credentials is done outside this component and thus part of the environment.

EXT_FIA_UAU.2 User authentication before any action

EXT_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.

EXT_FIA_UAU.2.2 The TOE shall initiate the verification of [*assignment: user credentials that can be used to authenticate S.USER*].

Note:

The verification of the user credentials is done in the Gateway Authentication process (see chapter 7.1 for more information)[3].

Unlike FIA_UAU.2 (component from CC part II) the required verification of the user credentials done outside this component and thus part of the environment.

EXT_FIA_UID.2 User identification before any action

EXT_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.

Note:

Unlike FIA_UID.2 (component from CC part II) the required verification of the user credentials done by local operating system or AD server is done outside this component and thus part of the environment.

Application note:

“other TSF-mediated actions” (EXT_FIA_UID.2 and EXT_FIA_UAU.2) means, that the user is now authorized to access the destined network resource which is represented by FTP_ITC.1.

6.1.3 Class FDP – User Data Protection

FDP_ACC.1 Subset access control

FDP_ACC.1.1 The TSF shall enforce the [*assignment*: *SFP.ACCESS*] on [*assignment*: *S.USER, O.NETWORK, R.CONNECT*].

FDP_ACF.1 Security attribute based access control

FDP_ACF.1.1 The TSF shall enforce the [*assignment: SFP.ACCESS*] to objects based on the following [*assignment: user credentials of S.USER*].

FDP_ACF.1.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [*assignment: S.USER is only allowed to R.CONNECT to O.NETWORK if it is identified and authenticated on behalf of its security attribute*].

FDP_ACF.1.3 The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: [*assignment: None*].

[3] UAG supports following user credentials: Username and password, Challenge/Response (NTLM), and Client Certificate. Username and password (Forms-based authentication) has been used for evaluation.

FDP_ACF.1.4 The TSF shall explicitly deny access of subjects to objects based on the [*assignment: None*].

6.1.4 Class FMT – Security Management

FMT_SMF.1 Specification of Management Functions

FMT_SMF.1.1 The TSF shall be capable of performing the following management functions: [*assignment:*
a) add, modify, delete security attributes according to SFP.ACCESS;
b) query audit data associated with SFP.ACCESS
c) set maximum size of audit data
].

FMT_MSA.1 Management of security attributes

FMT_MSA.1.1 The TSF shall enforce the [*assignment*: *SFP.ACCESS*] to restrict the ability to [*selection*: *add, query, modify, delete*] the security attributes [*assignment*: *attributes associated with S.USER, R.CONNECT*] to [assignment: *S.ADMIN*].

FMT_MSA.3 Static attribute initialisation

FMT_MSA.3.1 The TSF shall enforce the [*assignment: SFP.ACCESS*] to provide [*selection*: *restrictive*] default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2 The TSF shall allow the [*assignment: S.ADMIN*] to specify alternative initial values to override the default values when an object or information is created.

Application Note:

The TOE does not maintain the role S.ADMIN. Access control to the TOE is granted by the underlying operating system which also maintains the role S.ADMIN. So the environment fulfils FMT_SMR.1.

6.1.5 Class FTP – Trusted path/channels

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 [*selection:* 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 [assignment: R.CONNECT].

Application Note:

The trusted IT product mentioned in FTP_ITC.1 is the web browser of a user who connects to the TOE from the external network using HTTPS (SSL/TLS secured HTTP).

6.2 Security Assurance Requirements

The Security Assurance Requirements for the TOE are the assurance components of Evaluation Assurance Level 2 (EAL2) augmented with ALC_FLR.3 (printed in bold in the table below). They are all drawn from Part 3 of the Common Criteria. The assurance components are listed in Table 6.3.

Table 6.3 – EAL2 (augmented) Assurance Requirements

Assurance Class	**Assurance Component Documentation Requirements**
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
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.3: Systematic flaw remediation
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

6.3 Security Requirements Rationale

6.3.1 Rationale for the security functional requirements

The mapping of security objectives to functional requirements (components) is provided in Table 6.4.

Table 6.4 – Security Objective to Functional Component Mapping

Security Objectives vs. Functional Component	O.ACCOUN	O.AUDREC	O.IDAUTH	O.MEDIAT	O.SECSTA
EXT_FIA_AFL.1			X		
EXT_FIA_UAU.2	X		X		
EXT_FIA_UID.2	X		X		
FAU_GEN.1	X	X			
FAU_SAR.1		X			
FAU_SAR.3		X			
FAU_STG.3		X			
FDP_ACC.1				X	
FDP_ACF.1				X	
FMT_MSA.1				X	
FMT_MSA.3				X	X
FMT_SMF.1	X	X	X	X	X
FTP_ITC.1				X	

A discussion of the rationale for the mapping is provided for each security objective below.

O.ACCOUN: The TOE must provide user accountability for information flows through the TOE and for authorized administrator use of security functions related to audit.

O.ACCOUN is mapped to FAU_GEN.1, EXT_FIA_UID.2, EXT_FIA_UAU.2, FMT_SMF.1.

- FAU_GEN.1 Audit data generation

  This component outlines what data must be included in audit records. Audit data generated by the TOE is stored in different log files as stated in SF3. When applicable, information about S.USER is stored in the log files.

- EXT_FIA_UID.2 User identification before any action

  This component ensures that S.USER identify himself (when required) before any information is passed though the TOE.

- EXT_FIA_UAU.2 User authentication before any action

  This component ensures that S.USERs are identified when necessary. When authentication is required it must occur before any data is passed though the TOE. Note, S.ADMINs are not authenticated by the TOE itself. This is done by the environment (underlying operating system).

- FMT_SMF.1 Specification of Management Functions

  This component ensures that S.ADMIN is able to manage security attributes associated with SFP.ACCESS. So it enables adding, modifying, deleting security attributes according to SFP.ACCESS and the querying audit data associated with SFP.ACCESS.

O.AUDREC: The TOE must provide a means to record a readable audit trail of security-related events, with accurate dates and times, and a means to search and sort the audit trail based on relevant attributes. The TOE must provide that the audit trail is readable and no records are left because of not enough storage capacity.

O.AUDREC is mapped to FAU_GEN.1, FAU_SAR.1, FAU_SAR.3, FAU_STG.3, FMT_SMF.1.

- FAU_GEN.1 Audit data generation

  This component outlines what data must be included in audit records. Audit data generated by the TOE is stored in different log files as stated in SF3. When applicable, information about the identified user is stored in the log files.

- FAU_SAR.1 Audit review

  This component ensures that the user can interpret the recorded information. The log data is

  a) stored in a human readable form in files by the TOE and can be reviewed using the Web Monitor, or

  b) special events are stored in the Windows Event Log which can be reviewed with the Event Viewer (which is part of the operating system).

- FAU_SAR.3 Selectable Audit Review

  This component ensures that a variety of filtering, searching and sorting can be performed on the audit trail.

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

  This component ensures that the user is alerted in case of possible audit data loss.

- FMT_SMF.1 Specification of Management Functions

  This component ensures that S.ADMIN is able to manage security attributes associated with SFP.ACCESS. So it enables adding, modifying, deleting security attributes according to SFP.ACCESS and the querying audit data associated with SFP.ACCESS.

O.IDAUTH: The TOE must uniquely identify and authenticate the claimed identity of S.USER, before granting a user access to O.NETWORK.

O.IDAUTH is mapped to EXT_FIA_AFL.1, EXT_FIA_UID.2, EXT_FIA_UAU.2, FMT_SMF.1.

- EXT_FIA_AFL.1 Authentication failure handling

  This component exists to specify action after some number of unsuccessful authentication attempts. It ensures that S.USER cannot endlessly attempt to authenticate without leaving no trace in the log files.

- EXT_FIA_UID.2 User identification before any action

  This component ensures that S.USER identifies himself (when required) before any information is passed though the TOE.

- EXT_FIA_UAU.2 User authentication before any action

  This component ensures that S.USER are identified when necessary. When authentication is required it must occur before any data is passed though the TOE. Note, that S.ADMINs are not authenticated by the TOE itself. This is done by the environment (underlying operating system).

- FMT_SMF.1 Specification of Management Functions

  This component ensures that S.ADMIN is able to manage security attributes associated with SFP.ACCESS. So it enables adding, modifying, deleting security attributes according to SFP.ACCESS and the querying audit data associated with SFP.ACCESS.

O.MEDIAT: The TOE must mediate the flow of all information from S.USER on O.NETWORK.

O.MEDIAT is mapped to FDP_ACC.1, FDP_ACF.1, FMT_MSA.1, FMT_MSA.3, FMT_SMF.1, FTP_ITC.1.

- FDP_ACC.1 Subset access control, FDP_ACF.1 Security attribute based access control

  These components define the access control policy SFP.ACCESS on S.USER connecting (R.CONNECT) to O.NETWORK.

- FMT_MSA.1 Management of security attributes

  This component ensures that S.ADMIN is able to management the security attributes of SFP.ACCESS.

- FMT_MSA.3 Static attribute initialization

  This component ensures that there is a restrictive default policy for security attributes that are used to enforce SFP.ACCESS. The TOE ensures that by default no user is allowed to connect to O.NETWORK.

- FMT_SMF.1 Specification of Management Functions

  This component ensures that S.ADMIN is able to manage security attributes associated with SFP.ACCESS. So it enables adding, modifying, deleting security attributes according to SFP.ACCESS and the querying audit data associated with SFP.ACCESS.

- FTP_ITC.1 Inter-TSF trusted channel
  This component ensures that the TOE enforces a secure channel between user clients and the TOE using SSL/TLS, in order to protect the transmitted data.

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.SECSTA is mapped to FMT_MSA.3, FMT_SMF.1.

- FMT_MSA.3 Static attribute initialization

  This component ensures that there is a restrictive default policy for security attributes that are used to enforce SFP.ACCESS. The TOE ensures that by default no user is allowed to connect to O.NETWORK.
- FMT_SMF.1 Specification of Management Functions

  This component ensures that S.ADMIN is able to manage security attributes associated with SFP.ACCESS. So it enables adding, modifying, deleting security attributes according to SFP.ACCESS and the querying audit data associated with SFP.ACCESS.

6.3.2 Dependencies of security functional requirements

Table 6.5 – TOE Functional Requirements Dependencies

Requirement (SFR TOE)	**Dependencies**	**Dependency fulfilled**
FAU_GEN.1	FPT_STM.1	A.ENV
FAU_SAR.1	FAU_GEN.1	yes
FAU_SAR.3	FAU_SAR.1	yes
FAU_STG.3	FAU_STG.1	A.ENV
EXT_FIA_AFL.1	EXT_FIA_UAU.1	yes
EXT_FIA_UAU.2	EXT_FIA_UID.1	yes
EXT_FIA_UID.2	none	yes
FTP_ITC.1	none	yes
FDP_ACC.1	ADP_ACF.1	yes
FDP_ACF.1	FDP_ACC.1, FMT_MSA.3	yes yes
FMT_MSA.1	[FDP_ACC.1, or FDP_IFC.1], FMT_SMR.1, FMT_SMF.1	Yes, by FDP_ACC.1 A.ENV yes
FMT_MSA.3	FMT_MSA.1 FMT_SMR.1,	yes A.ENV
FMT_SMF.1	none	yes

All TOE Functional Requirements Dependencies are either fulfilled by the TOE Functional Requirement hierarchy, by a TOE SFR, or by the IT environment.

The timestamp is provided by the underlying operating system. So FPT_STM.1 is related to A.ENV.

FAU_STG.1 is provided by the environment, as the TOE uses file I/O functionality of Windows Server.

The TOE does not maintain the management of user roles which is done by the underlying operating system. So FMT_SMR.1 is related to A.ENV.

6.3.3 Rationale for the assurance requirements

EAL2 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. In future, the TOE will also head for an EAL4 evaluation, as it is supposed to be resistant even against enhanced-basic attack potentials.

EAL2 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 has been analyzed; the augmentation with ALC_FLR.3 ensures that the developer has documented a systematic flaw remediation procedure, that describe the procedures used to track all reported security flaws, the status of finding a correction of the flaw and the methods used to provide flaw information, corrections and guidance on corrective actions, provide a flaw remediation procedure, a procedures for processing reported security flaws, and a flaw remediation guidance. 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, evidence of a developer search for vulnerabilities, and a vulnerability analysis demonstrating resistance to penetration attackers with a basic attack potential.

Beside this general description, the TOE itself acts as application level gateway with a basic level of protection and should be used in application scenarios where basic level of protection is sufficient.

7 TOE Summary Specification

The TOE summary specification in the following specifies the security functionality in form of security functions as well as the assurance measures of the TOE.

The TOE consists of four security functionalities (SF) which will be described in more detail in the following chapters. These security functionalities are:

SF1: Access Control
describes the access control mechanism for S.USER

SF2: Information Protection
describes R.CONNECT of S.USER to O.NETWORK

SF3: Audit
describes the audit capabilities

SF4: Management
describes the management capabilities.

7.1 SF1 - Access Control

UAG allows to control client endpoint access to published resources. Forefront UAG allows to do the following:

- Requires an HTTPS channel between client endpoints and the Forefront UAG server.
- Apply session authentication. You can require client endpoints to authenticate in order to connect to a portal.
- Require client endpoints to be authorized based on ACLs in order to access published applications in a portal

Relevant SFRs for this SF are: EXT_FIA_AFL.1, EXT_FIA_UAU.2, EXT_FIA_UID.2, FDP_ACC.1, FDP_ACF.1, FMT_MSA.3

7.1.1 About session authentication

Forefront UAG enables you to control access to internal resources by checking users against an authentication database. A portal or application session is opened only for users who authenticate successfully. Users who cannot authenticate successfully do not gain access. Access is granted per user, and each authentication instance is only valid for one connection. Forefront UAG can seamlessly integrate with numerous authentication schemes, even if the application being protected has no inherent support for the method you choose to implement, where Forefront UAG serves as a client of the third-party authentication server. In addition, Forefront UAG also enables non-intrusive, forced, periodic re-authentication by applying a logoff scheme. After a predetermined time, users must re-enter credentials to continue working. If they do, they resume working where they left off; if they do not, their sessions are terminated.

To define session authentication you should define an authentication server against which credentials of users connecting to a portal or application session are verified.

Access control lists are used to specify which authorized user is allowed to access a certain application.

The verification of S.USER is done in the environment. The process is initiated and finished by the TOE.

7.1.2 Client authentication Servers

Forefront Unified Access Gateway (UAG) allows you to configure the authentication scheme for applications that require authentication.

Forefront UAG supports many types of authentication servers:

Authentication Servers supported	**Evaluated**
Configuring Active Directory authentication	yes
Configuring LDAP authentication	yes
Configuring RADIUS authentication	no
Configuring ACE authentication	no
Configuring TACACS authentication	no
Configuring WinHTTP authentication	no
Configuring NT Domain authentication	no
Configuring Notes Directory authentication	no
Configuring Novell Directory authentication	no
Configuring custom authentication	no

7.2 SF2 – Information Protection

Using UAG you can make internal applications and resources available to remote endpoints by publishing them in an application Web site or portal. Portals and applications are published using a Forefront UAG transfer channel known as a trunk. Remote endpoints connect to Forefront UAG trunks over HTTPS. You can publish two kinds of trunks:

- Portal trunk
  A one-to-many connection with a single IP address used to access multiple applications published in the portal.
- ADFS trunk
  Publishing of an Active Directory Federation Server for session authentication (not in scope of the evaluation)

When you create a trunk to publish a portal application, you can specify that client endpoints communicate with the Forefront UAG server over HTTPS connection. In this case, you must select a server certificate when you configure the trunk. This certificate is used to authenticate the Forefront UAG server to the client endpoint.

Relevant SFRs for this SF are: FTP_ITC.1

7.2.1 Portal trunks

You can use a portal trunk to publish applications and internal file structures, and to enable VPN access to the internal network. Using a portal you can publish the following:

- **Web applications**
  Web applications that use HTTPS and a Web interface.
- **Built-in services (not evaluated)**
  Built-in services are supplied by Forefront UAG, such as File Access or Web Monitor.
- **Client/server and legacy applications (not evaluated)**
  Client/server and legacy applications use non-HTTP or non-HTTPS protocols.
- **Browser-embedded applications (not evaluated)**
  Browser applications are Web-initiated and use a Web-based interface to create a non-HTTP or non-HTTPS connection.

Portal trunks provide the following advantages:

- Provide a single point of entry to all of your published applications.
- Allow you to add or remove published applications depending on your requirements.

7.2.2 Application publishing

You can publish several types of applications in a Forefront Unified Access Gateway (UAG) portal trunk. For each application, the publishing process is as follows:

1. Select an application.
2. Specify an application name and type.
3. Select an access policy for the application.
4. Select whether to publish a single Web application or a Web farm (not applicable to all applications, not part of the evaluation).
5. Configure server settings.
6. Configure how to authenticate users to the server.
7. Configure the portal link.
8. Configure which users are authorized for the application.

After publishing your applications, you can configure many of the settings of the published application, including:

- The application name and any prerequisite applications that may be required for the application.
- The application server address and port.
- Content inspection for application traffic (not evaluated).

- Client endpoint policies (not evaluated).

Predefined Web Application
Microsoft Dynamics CRM 4.0
Microsoft Exchange Server
Microsoft Forefront Identity Manager 2010
Microsoft Office Communicator Web Access 2007
Microsoft Office SharePoint Portal Server 2003
Microsoft Office SharePoint Server 2007
Microsoft SharePoint Server 2010 (Beta)

7.2.3 Publishing Web servers

When publishing Web applications on UAG, you can choose to publish the application as a single Web server or as a Web farm (the Web farm feature has not been evaluated and is mentioned for completeness only).

- Applications published as a single Web application have only one application server within the internal network. The application server must be sized appropriately for the predicted number of users that will access this application.
- Applications published as a Web farm have more than one application server within the internal network. When configuring this application in the trunk, you define the IP addresses or host names of each application server. You also define the load balancing method to use for requests to this application.

7.3 SF3 - Audit

The TOE stores logging information in the environment:

- UAG event log

  The UAG log contains records of system, security, session and application events. These events are logged in a file in the environment. Each log entry has a time stamp.
- Windows application event log

  The Windows application event log stores important system events and failures.

and detects the occurrence of the following selected events:

- configuration changes (system event),
- access control (session event),
- failed authentication of users (session event),
- trusted channel established (session event),

- service started, stopped or not responding, which implies startup and shutdown of audit functions (Windows event log).
- disk full, which implies the audit log full event (Windows event log)

The log files can be audited[4] using the Web Monitor of the TOE. The Windows event log can be reviewed with the Event Viewer (which is part of the operating system).

Note:

The TOE provides the ability to perform filter, search and sort operations on the recorded audit data using the “event query” function. The selected, found or sorted data is displayed using the Web Monitor.

Relevant SFRs for this SF are: FAU_GEN.1, FAU_SAR.1, FAU_SAR.3, FAU_STG.3

7.4 SF4 - Management

UAG provides management of the TOE through the Forefront UAG Management. The GUI exposes the main administration functionality necessary for the administering the TOE.

UAG management configuration data is stored is stored in ADAM. Ability to read and modify those objects and attributes is controlled through the file systems’ access control lists (by the environment).

Further, the TOE provides a command-line utility to manage a few logging settings.

Relevant SFRs for this SF are: FMT.MSA.1, FMT_SMF.1

SF4 will specifically provide the following management functionality for the other security functionalities:

7.4.1 Management for Access Control

The TOE provides functionality to manage the Identification and Authentication of the frontend authentication of S.USER.

The frontend authentication data is forwarded to environment authentication servers which can be configured using Forefront UAG Management.

The TOE further provides functionality to

- Manage Session Authentication
- Manage Authentication Servers
- Manage ACLs for user access to applications

[4] This includes several sorting and filtering features.

7.4.2 Management for Information Protection

The TOE provides functionality to configure whether a secured channel (HTTPS) shall be used for a trunk or not (HTTP).

7.4.3 Management for Audit

The TOE provides functionality to

- set the audit size limit before older audit files are deleted

7.5 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 7.1 – Assignment of security functional requirements to security functions

SFR	SF1	SF2	SF3	SF4
EXT_FIA_AFL.1	X			
EXT_FIA_UAU.2	X			
EXT_FIA_UID.2	X			
FAU_GEN.1			X	
FAU_SAR.1			X	
FAU_SAR.3			X	
FAU_STG.3			X	
FDP_ACC.1	X			
FDP_ACF.1	X			
FMT_MSA.1				X
FMT_MSA.3	X			
FMT_SMF.1				X
FTP_ITC.1		X		

EXT_FIA_AFL.1 (Authentication failure handling) is mapped to SF1. This component exists to specify action after some number of unsuccessful authentication attempts. It ensures that S.USER cannot endlessly attempt to authenticate without leaving no trace in the log files.

EXT_FIA_UAU.2 (User authentication before any action) is mapped to SF1 and ensures that S.USERs are identified when necessary. When authentication is required it must occur before any data is passed though the TOE.

Note, that S.ADMIN are not authenticated by the TOE itself. This is done by the environment (underlying operating system).

EXT_FIA_UID.2 (User identification before any action) is mapped to SF1. This component ensures that S.USER identifies himself (when required) before any information is passed though the TOE.

FTP_ITC.1 (Inter-TSF trusted channel) is mapped to SF2. This component represents the VPN tunnel which ensures that user data send or received by S.USER is protected.

FDP_ACC.1 (Subset access control) is mapped to SF1. This component enforces the SFP.ACCESS on S.USER, O.NETWORK and R.CONNECTION.

FDP_ACF.1 (Security attribute based access control) is mapped to SF1. It enfoces the SFP.ACCESS to objects based on security attributes of S.USER and ensures that S.USER is only allowed to R.CONNECT to O.NETWORK if it is identified and authenticated on behalf of its security attributes.

FMT_SMF.1 (Specification of Management Functions) is mapped to SF4. This component ensures that S.ADMIN is able to management the security attributes of SFP.ACCESS.

FMT_MSA.1 (Management of security attributes) is mapped to SF4. This component ensures that S.ADMIN is able to manage security attributes associated with SFP.ACCESS. So it enables adding, modifying, deleting security attributes according to SFP.ACCESS and the querying audit data associated with SFP.ACCESS.

FMT_MSA.3 (Static attribute initialization) is mapped to SF1. This component ensures that there is a restrictive default policy for the VPN tunnel.

FAU_GEN.1 (Audit data generation) is mapped to SF3 and outlines what data must be included in audit records. Audit data generated by the TOE is stored in different log files as stated in SF3. When applicable, information about the identified user is stored in the log files.

FAU_SAR.1 (Audit review) is mapped to SF3 and ensures that the user can interpret the recorded information. The log data is

a. stored in a human readable form in files by the TOE and can be reviewed using the Web Monitor, or
b. special events are stored in the Windows Event Log which can be reviewed with the Event Viewer (which is part of the operating system).

FAU_SAR.3 (Selectable Audit review) is mapped to SF3 and ensures that a variety of filtering, searching and sorting can be performed on the audit trail.

FAU_STG.3 (Action in case of possible audit data loss) is mapped to SF3 and ensures that S.ADMIN is alerted in case of possible audit data loss.

8 Appendix

8.1 References

[CC] *Common Criteria for Information Technology Security Evaluation*, version 3.1, revision 3, July 2009
Part 1: Introduction and general model, CCMB-2009-07-001,
Part 2: Security functional requirements, CCMB-2009-07-002,
Part 3: Security Assurance Requirements, CCMB-2009-07-003

[MSUAG] *Microsoft Forefront UAG documentation,* Microsoft Corp.

[MSUAG_ADD] *Microsoft Forefront UAG Common Criteria Evaluation - Guidance Documentation Addendum,* Microsoft Corp.

[WEBUAG] Website: *Microsoft Forefront UAG - Common Criteria Evaluation*, https://go.microsoft.com/fwlink/?LinkId=210419

8.2 Acronyms

ACL	Access Control List
ADAM	Active Directory Application Mode
API	Application Programming Interface
CC	Common Criteria
EAL	Evaluation Assurance Level
GUI	Graphical User Interface
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
IIS	Internet Information Server/Services
IT	Information Technology
LDAP	Lightweight Directory Access Protocol
MSDN	Microsoft Developer Network
NIC	Network Interface Card
NTLM	NT LAN Manager
PP	Protection Profile
SF	Security Functionality
SFP	Security Function Policy
SFR	Security Functional Requirement
SSL	Secure Socket Layer

ST	Security Target
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Functionality
TSP	TOE Security Policy
VPN	Virtual Private Network

8.3 Glossary

Active Directory
Active Directory is a so called Directory Service. It promises to support a single unified view of objects on a network and allows locating and managing resources faster and easier.

ADAM
ADAM is a Lightweight Directory Access Protocol (LDAP) directory service that runs as a user service, rather than as a system service.

Active Directory Application Mode (ADAM) is a part of Microsoft’s fully integrated directory services available with Windows Server, and is built specifically to address directory-enabled application scenarios. ADAM runs as a non-operating-system service, and, as such, it does not require deployment on a domain controller. Running as a non-operating-system service means that multiple instances of ADAM can run concurrently on a single server, and each instance can be configured independently.

ADAM Configuration Receiver
The configuration is replicated from ADAM to the registry and file system by a service called ADAM Configuration Receiver Service.

authentication
Authentication is "A positive identification, with a degree of certainty sufficient for permitting certain rights or privileges to the person or thing positively identified." In simpler terms, it is "The act of verifying the claimed identity of an individual, station or originator" [Schou, Corey (1996). Handbook of INFOSEC Terms, Version 2.0. CD-ROM (Idaho State University & Information Systems Security Organization)]

credentials
An authentication method used to validate client-to-server and server-to-server communication. Credentials include a user name and a password that is used to validate requests from client computers or from other computers in an array or chain.

DirectAccess
Forefront UAG DirectAccess provides remote users with the experience of a seamless connection to your internal network any time that you have Internet access. When Forefront UAG DirectAccess is enabled, requests for internal network resources (such as e-mail servers, shared folders, management servers, or intranet Web sites) are securely directed to the internal network, without the need to connect to a VPN.

Form-based authentication
Form-based authentication is a method of authenticating users using web-based forms for providing credentials.

Identification
Identification, according to a current compilation of information security terms, is "the process that enables recognition of a user described to an automated data processing system. This is generally by the use of unique machine-readable names" [Schou, Corey (1996). Handbook of INFOSEC

	Terms, Version 2.0. CD-ROM (Idaho State University & Information Systems Security Organization)].
Integrated Windows authentication	formerly named NTLM or Windows NT Challenge/Response authentication
Network interface card (NIC)	A NIC or Network Interface Card is a circuit board or chip, which allows the computer to communicate to other computers on a Network.
Secure Sockets Layer (SSL)	A protocol that supplies secure data communication through data encryption and decryption. SSL enables communications privacy over networks.
TLS	Transport Layer Security: TLS is based on the SSL 3.0 Protocol Specification
TMG	Threat Management Gateway
Trunk	A trunk is a transfer channel that is used to access published applications. A trunk is always associated with a TCP port where user clients (web browsers) are supposed to connect to. Trunks contain a lists of applications that can be accessed by connecting to it.
UAG	Unified Access Gateway
WinAPI	Windows API which is used by user applications to access Windows functionality for file I/O, memory management, registry access, and other basic functionality that operating systems usually provide.