Windows Server® 2008 R2
Active Directory® Federation Services 2.0

Active Directory Federation Services 2.0

Security Target

Common Criteria: EAL4 augmented with ALC_FLR.3

Version 1.0

23-MAR-12

Document history

Version	Date	Description
0.1	10-MAY-11	Initial draft for review.
0.2	27-MAY-11	Updated to address evaluator comments for submission.
0.3	30-NOV-11	Updated to address EOR001
1.0 (Draft)	24-MAR-12	Updated TOE version

Table of Contents

1 Security Target Introduction (ASE_INT)

1.1 Background

Employees require security-enhanced access to a growing number of on-premises applications, cloud services, and other resources. Organizations want that access to be easy, yet flexible enough to accommodate collaboration across organizational boundaries. Access must comply with internal security policies and external regulations. In addition, organizations need to readily adapt to changing business needs and technology trends, such as the emergence of more hosted services and service-oriented architecture models.

Today, few organizations have successfully implemented such a comprehensive solution. The root of the problem is that applications rely on custom access control logic which is dependent on existing IT infrastructure and methods that do not provide a collaborative approach to managing identity and authorization to resources.

Active Directory Federation Service (AD FS) solves this problem by enabling the federation of identity and access management by securely sharing digital identity and entitlements rights across security and enterprise boundaries. AD FS extends the ability to use single sign-on functionality that is available within a single security or enterprise boundary to Internet-facing applications to enable customers, partners, and suppliers a streamlined user experience while accessing the web-based applications of an organization.

AD FS supports open standards such as the Security Assertion Markup Language (SAML), an XML-based open standard for exchanging authentication and authorization data between security domains. AD FS also supports a range of other client authentication methods such as Kerberos, X.509 and user name/password. AD FS also interacts with a range of user identity and attribute stores. This flexibility allows AD FS to co-exist with existing Windows security capabilities and other external trust infrastructure.

AD FS helps IT enable users to collaborate across organizational boundaries and easily access applications on-premise and in the cloud, while maintaining application security by:

a) establishing, managing and distributing an identity claims ruleset;

b) establishing and managing trust relationships between independent organizations; and

c) managing access requests by establishing, creating, transforming and distributing authentication and authorization claims through a standard security token.

1.2 ST reference

ST Title	Microsoft Active Directory Federation Services 2.0 Security Target
ST Version/Date	1.0 (Draft) (23-MAR-12)
TOE Reference	Microsoft Active Directory Federation Services 2.0, which includes the following: a) Federation Service (Build KB974408) b) Federation Service Proxy (Build KB974408) c) AD FS Management Console (Build KB974408)
CC Identification	Common Criteria for Information Technology (IT) Security Evaluation, Version 3.1 (REV 3) July 2009, incorporating: a) Part One – Introduction and General Model, b) Part Two – Security Functional Components, and c) Part Three – Security Assurance Component.

1.3 Document organization

This document is organized into the following major sections:

a) Section 1 provides the introductory material for the ST and the TOE overview (ASE_INT).

b) Section 2 provides the conformance claims for the evaluation (ASE_CCL).

c) Section 3 provides the definition of the security problem addressed by the TOE (ASE_SPD).

d) Section 4 defines the security objectives for the TOE and the environment (ASE_OBJ).

e) Section 5 contains the security functional requirements derived from the Common Criteria, Part 2 (ASE_REQ).

f) Section 6 contains the security assurance requirements derived from the Common Criteria, Part 3 (ASE_REQ).

g) Section 7 provides the TOE summary specification that demonstrates how the TOE implements the claimed security functions.

h) Annex A provides defined terms (ASE_REQ).

i) Annex B provides a suite of correspondence mappings and required rationale.

1.4 TOE overview

1.4.1 TOE type

The target of evaluation (TOE) is an identity access solution providing a suite of software components that manage and process authentication and authorization claims across trusted organisational network boundaries and also across heterogeneous environments. Fundamentally the TOE provides the necessary infrastructure for implementing a web-based single sign on (SSO) capability for claims-aware applications for both local users and external users from trusted partner organisations.

The TOE can be categorised in the ***access control devices and systems*** category in accordance with those categories identified on the Common Criteria Portal that lists all certified products.

1.4.2 TOE usage and major security features

The TOE is generally used in an enterprise environment and provides browser-based clients with a seamless single-sign on (SSO) experience across multiple Internet-facing web-based applications that are considered claims-aware applications. AD FS helps simplify the user experience by enabling the issue of a single token that can be used over multiple related web applications over the life of single online session. AD FS can be used to provide federated access to multiple claims-aware applications for users in partner and trusted organisations.

The TOE is deployed in one of the following three (3) specific ways to support the identity and access goals of an organization:

a) **Providing your organization's Active Directory users with access to corporate claims-aware applications and services.** Users in your organization can access an AD FS 2.0–secured application or service (either your organization's application or service or a partner's application or service) when the users are logged on to Active Directory in the corporate intranet. In this deployment model:

    i. Corporate network users who are logged on to an Active Directory forest in the corporate network can use SSO to access multiple applications or services in the organization's perimeter network.

    ii. Remote users who are logged on to an Active Directory domain can obtain AD FS tokens from the federation server to gain federated access to AD FS-secured web-based applications or services that also reside within the organization.

    iii. Information in the Active Directory attribute store can be populated into an employee's AD FS security token.

b) **Providing your organization's Active Directory users with access to the applications and services of other organizations.** Users in your organization can access an AD FS 2.0–secured application or service (either your organization's own application or service or a partner's application or service) when the users are logged on to an attribute store in the corporate intranet and when they log on remotely from the Internet. The administrator in the account partner organization has a deployment goal to provide federated access to partner organization users. In this deployment model:

   i. Corporate network users who are logged on to an Active Directory domain in the corporate network can use SSO functionality to access multiple web-based applications or services, which are secured by AD FS 2.0, when the applications or services are in a different organization.

   ii. Remote employees who are logged on to an Active Directory domain can obtain AD FS 2.0 tokens from the federation server in your organization to gain federated access to AD FS 2.0–secured web-based applications or services that are hosted in another organization.

c) **Provide federated users in another organization access to your claims-aware applications and services.** User accounts in another organization that are located in an attribute store on that organization's corporate network require access an AD FS 2.0–secured application in your organization. This goal also works when consumer-based user accounts that are located in an attribute store in your organization's perimeter network must be provided with access to an AD FS 2.0–secured application in your organization. In this deployment model:

   i. Federated users both in your organization and in organizations who have configured a federation trust to your organization (account partner organizations) can access the AD FS secured application or service that is hosted by your organization.

   ii. Federated users who have no direct association with a trusted organization (such as individual customers), who are logged on to an attribute store that is hosted in your perimeter network, can access multiple AD FS-secured applications, which are also hosted in your perimeter network, by logging on one time from client computers that are located on the Internet.

There are two (2) sides in any given federation relationship, one side supplies the users with the requested claims while the other side relies on the presented claims to authorize access to supplied applications or resources. The ***claims provider*** is responsible for collecting and authenticating a user's credentials, building up claims for that user, and packaging the claims into security tokens for issuance to the user. Whereas the ***relying party,*** the second organizational partner in the federation trust relationship, trusts the account partner to authenticate users and verifies that security token provided by users are really issued by the trusted account partner.

The following table describes the security features that are central to the security functional requirements that are being claimed for this evaluation of AD FS.

Security features	Descriptions
Claims management	AD FS provides a claims rules engine that is used to manage claims policy for the implementation of the capability. In the context of digital identities, claims are statements that one subject makes about itself or another subject. These claims can be made by a person directly or provided to others by a third party. Other parties can rely on the values of the claims to perform a process of digital identification.
Trust management	AD FS implements the capability to manage cross-organizational (federation-based) collaboration. The determination of the trust relationship depends on whether the organisation will host a web resource to be accessed by other organizations across the Internet—or the reverse. Therefore, there are a two types of partners in the relationship that is to be managed by AD FS: • **Account partner.** An account partner represents the organization in the federation trust relationship that physically stores user accounts in either an Active Directory Domain Services (AD DS) store or an Active Directory Lightweight Directory Services (AD LDS) store. The account partner is responsible for collecting and authenticating a user's credentials, building up claims for that user, and packaging the claims into security tokens. These tokens can then be presented across a federation trust for access to resources that are located in the resource partner organization. • **Resource partner.** A resource partner is the second organizational partner in the federation trust relationship. A resource partner is the organization where the AD FS-enabled web servers that host one or more web-based applications. The resource partner trusts the account partner to authenticate users. Therefore, to make authorization decisions, the resource partner consumes the claims that are packaged in security tokens coming from users in the account partner.
Token issuance	AD FS implements a security token service (STS) that processes all claims and requests for tokens using the following steps: • **Step 1: Accept incoming claims.** Extract the incoming claims from the token and eliminate claims that are not expected or trusted. After they are extracted, the acceptance rules that make up the acceptance transform rule set for a claims provider trust are run. • **Step 2: Authorize the claims requester.** This stage is used by the claims engine to issue permit or deny claims based on whether the token requester is allowed to obtain a token for the given relying party or not. • **Step 3: Issue outgoing claims.** This stage is used to issue outgoing claims and send them along the pipeline where they will be packaged into a security token.

1.4.3 Supporting hardware

The following minimum and recommended hardware requirements apply to the servers that host the Federation Server and the Federation Service Proxy components.

Hardware	Minimum requirement	Recommended requirement
CPU speed	Single-core, 1 gigahertz (GHz)	Quad-core, 2 GHz
RAM	1 GB	4 GB
Disk space	50 MB	100 MB

1.4.4 Supporting software

For a base installation platform, AD FS requires either the Windows Server 2008 or the Windows Server 2008 R2 operating system. During the AD FS installation process, the setup wizard attempts to automatically check for and, if necessary, install the following prerequisite applications and hotfixes:

a) Windows Hotfix (KB968389) - Installed only on Windows Server 2008 computers

b) Windows Hotfix (KB970430) - Installed only on Windows Server 2008 computers

c) Windows Hotfix (KB973917) - Installed only on Windows Server 2008 computers

d) Windows Hotfix (KB975955) - Installed only on Windows Server 2008 computers

e) Windows Hotfix (KB981002) - Installed only on Windows Server 2008 R2 computers

f) Windows Hotfix (KB981201) - Installed only on Windows Server 2008 computers

g) Windows Hotfix (KB981202) - Installed only on Windows Server 2008 computers

h) Windows Hotfix (KB981205) - Installed only on Windows Server 2008 computers

i) Microsoft .NET Framework 3.5 Service Pack 1 (SP1) - Installed only on Windows Server 2008 R2 computers

j) Internet Information Services (IIS) 7

k) Windows Identity Foundation (WIF)

l) Windows PowerShell

1.4.5 Certificate requirements

Certificates play an important role in securing communications between federation servers, federation server proxies, claims-aware applications, and web clients. The requirements for certificates vary, depending on the server or client on which the certificate is being installed. The following table identifies the required certificates.

Server	Certificate type	Description
Federation server	Server authentication certificate (also referred to as a Service Communication Certificate in the AD FS 2.0 Management snap-in)	This is a standard Secure Sockets Layer (SSL) certificate that is used for securing communications between federation servers, clients, and federation server proxy computers.
	Token-signing certificate	This is a standard X509 certificate that is used for securely signing all tokens that the federation server issues.
	Token-decryption certificate	This is a standard SSL certificate that is used to decrypt any incoming tokens that are encrypted by a partner federation server. It is also published in federation metadata.
Federation server proxy	Server authentication certificate	This is a standard Secure Sockets Layer (SSL) certificate that is used for securing communications between a federation server proxy and Internet client computers.

1.4.6 Client requirements

Although any current web browser with JavaScript capability can be made to work as an AD FS client, the Web pages that are provided by default have been tested only against Internet Explorer versions 7.0 and 8.0, Mozilla Firefox 3.0, and Safari 3.1 on Windows. JavaScript must be enabled, and cookies must be enabled for browser-based sign-in and sign-out to work correctly.

Cookies that are used for authentication are always Secure Hypertext Transfer Protocol (HTTPS) session cookies that are written for the originating server. If the client browser is not configured to allow these cookies, AD FS cannot function correctly. Persistent cookies are used to preserve user selection of the claims provider.

1.4.7 Authentication requirements

AD FS integrates naturally with existing Windows authentication, such as Kerberos authentication, NTLM, smart cards, and X.509 v3 client-side certificates. Federation servers use standard Kerberos authentication to authenticate a user against a domain. Clients can authenticate by using forms-based authentication, smart card authentication, and Windows Integrated authentication.

The AD FS federation server proxy role makes possible a scenario in which the user authenticates externally using SSL client authentication. The federation server role can be configured to require SSL client authentication.

Although AD FS can enforce the type of credentials that it uses for authentication (passwords, SSL client authentication, or Windows Integrated authentication), it does not directly enforce authentication with smart cards. Therefore, AD FS does not provide a client-side user interface (UI) to obtain smart-card personal identification number (PIN) credentials.

1.5 TOE description

1.5.1 Physical scope of the TOE

Physically the TOE is an enterprise software application that is an optional server role in the Microsoft Windows Server 2008 operating system. Figure 1 identifies the various components that both comprise and support the AD FS solution.

This diagram presents a simplistic view of the deployment of the AD FS solution with the aim of identifying all of the major components that are considered within the scope of the evaluation. In reality implementation of an enterprise AD FS solution must consider many environment conditions and federation requirements to ensure that a secure and suitable capability is deployed according to the needs of the customer.

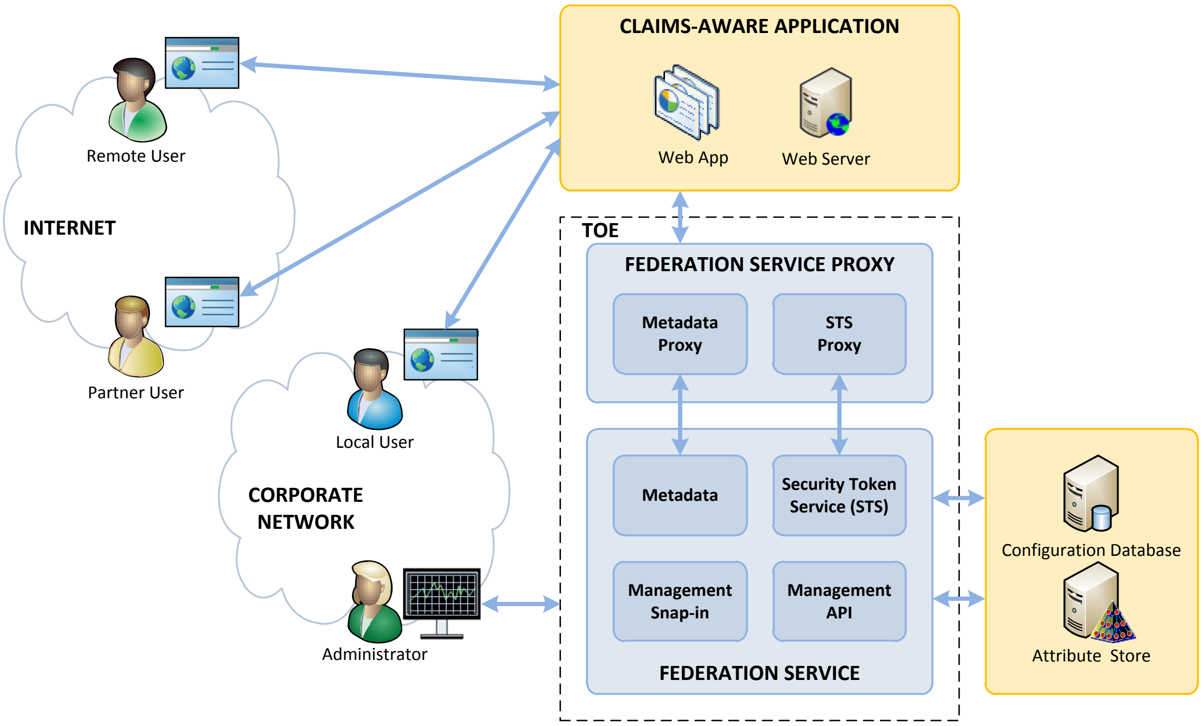

Figure 1 – TOE high-level architecture

As identified in Figure 1 above the following components comprise the AD FS solution:

a) **Federation service.** This component manages federation trust relationships between organizations and associated policies. The FS also issues security tokens for users successfully authenticated by the external attribute store that includes the claims data. The Federation Service can be deployed on one or more Federation Servers that share a common trust policy. The Federation Service routes and manages authentication requests and generates security tokens for local, remote business partner users from trusted organizations. The Federation Service is the core component of the capability housing the metadata and policy and also the security token service for the TOE.

b) **Federation service proxy.** The Federation Service Proxy component is proxy to the Federation Service in the perimeter network (also commonly known as a demilitarized zone or a screened subnet). The Federation Service Proxy uses WS-Federation Passive Requestor Profile (WS-F PRP) protocols to collect user credential information from browser clients, and it sends the user credential information to the Federation Service on the requesting user's behalf. This component processes client token requests and provides a user interface for browser-based clients.

c) **AD FS management snap-in.** The AD FS snap-in is a single Microsoft Management Console (MMC) snap-in. It provides a graphical user interface (GUI) for configuring service and policy settings that are used most commonly with AD FS solution.

The following components are considered outside the physical scope of the TOE, but are necessary software elements that support the TOE in delivering the security objectives:

a) **Attribute Stores.** AD FS uses the term ***attribute stores*** to refer to directories or databases that an organization uses to store its user accounts and their associated attribute values. After it is configured as an identity provider organization, AD FS retrieves these attribute values from the store and creates claims based on that information so that a web application or service that is hosted in a relying party organization can make the appropriate authorization decisions whenever a federated user (a user whose account is stored in the identity provider organization) attempts to access the application or service. The attribute store also provides the capability to authenticate the user's claims so that appropriate claims can be included in the returned token.

b) **Configuration database.** The AD FS configuration database stores all the configuration data that represents a single instance of AD FS (the Federation Service). The AD FS configuration database defines the set of parameters and rules that a Federation Service requires to identify partners, certificates, attribute stores, claims, and various data about these associated entities. This data store can be in either a Microsoft SQL Server database or the Windows Internal Database feature that is included with Windows Server 2008 and Windows Server 2008 R2.

c) **Claims-aware applications.** Developers will typically use the Windows Identity Foundation (WIF) to build a claims-aware application. However, while some elements of WIF have been used to develop the TOE, the use of this framework and development of the client-side applications is outside the scope of the evaluation.

1.5.2 Logical scope of the TOE

The logical scope of the TOE and the evaluation is centred on the core security functional policies and claims management activities as follows:

a) **Claims policy management.** Establishing, managing and distributing claims management policy.

b) **Trust management.** Establishing trust relationships between independent organizations and/or entities.

c) **Token issuance.** Managing authentication requests from various sources by establishing, creating, transforming and distributing authentication and authorization claims through a standard security token.

2 Conformance Claim (ASE_CCL)

This ST is conformant to version **3.1 (Rev 3)** of the Common Criteria for Information Technology Security Evaluation.

The following specific conformance claims are made for this ST:

a) **Part 2 conformant.** Conformant with Common Criteria for Information Technology Security Evaluation Part 2: Security functional requirements, version 3.1 (Rev 3).

b) **Part 3 conformant, EAL4 augmented with ALC_FLR.3.** Conformant with Common Criteria for Information Technology Security Evaluation Part 3: Security assurance requirements, version 3.1 (Rev 3).

3 Security problem definition (ASE_SPD)

3.1 Overview

This section describes the nature of the security problem that the TOE is designed to address. The security problem is described through:

a) a set of ***threats*** that the TOE must mitigate,

b) specific ***assumptions*** about the security aspects of the environment (both IT related and non-IT related elements) in which the TOE will operate, and

c) relevant ***organisational security policies*** that specify rules or guidelines that must be followed by the TOE and/or the operational environment.

3.2 Threats

In the context of this ST, the TOE has the following threat agents:

a) Individuals that have not been granted access to the TOE who attempt to gain access to information or functions provided by the TOE. This threat agent is considered an ***unauthorised individual***.

b) Individuals that are registered and have been explicitly granted access to the TOE who may attempt to access information or functions that they are not permitted to access. This threat agent is considered an ***unauthorised user***.

Identifier	Threat statement
T.AUTHENTICITY	An unauthorized user may attempt to forge a security token to gain access to protected resources resulting in unauthorized access to protected resources.
T.CONFIDENTIALITY	An unauthorized individual may access and modify configuration data (TSF data) that has been exported outside the scope of control of the TOE in an insecure manner resulting in loss of confidentiality of the data.
T.EXPORT	An unauthorized individual may access and modify claims data (user data) that has been exported outside the scope of control of the TOE in an insecure manner resulting in loss of integrity and confidentiality of the claims data.
T.IMPORT	An unauthorized individual may access and modify claims data (user data) that has been imported from outside the scope of control of the TOE in an insecure manner resulting in loss of integrity of the claims data.

Identifier	Threat statement
T.INTEGRITY	An unauthorized individual may access and modify configuration data (TSF data) that has been exported outside the scope of control of the resulting in loss of integrity of the data.
T.INTERNAL	An unauthorized individual may access and modify configuration data (TSF data) transmitted between physically separate parts of the TOE resulting in loss of confidentiality and integrity of the data.
T.TRUST	An unauthorized individual may access and modify trust relationship data (TSF data) transmitted between the claims provider and relying parties resulting in loss of confidentiality and integrity of the data.

3.3 Organisational security policies

In the context of this ST, the following organisational security policies (OSPs) are used to provide the basis for security objectives that are most often desired by acquirers and users of the TOE.

Identifier	OSP statement
P.ADMIN	Administrators must be capable of managing the full range of security functions and policies of the TOE.
P.BINDING	All authenticated users will be bound to provisioned security tokens to provide them with federated and SSO access to controlled resources.
P.AUTHENTICATE	All users requesting claims must be authenticated by an approved attribute store prior to processing claims and generating a security token.
P.DEFAULT	All identity resource access policies and authorization rules must be restrictive by default.
P.MANAGE	Administrators must be provided with an identity management tool supported by the TOE, and be capable of using it to manage the identities of TOE users.
P.PATH	All remote and federated users must be provided with a trusted interface for authenticating to the TOE to ensure the confidentiality and integrity of user and TSF data.
P.RESTRICTED	All administrative functions of the TOE must be restricted to the administrator.
P.ROLES	The TOE must be capable of associating all authenticated user with a specific role that relates to their location and expected interactions with the TOE.

Identifier	OSP statement
P.TOKEN	The TOE must be capable of applying administrator configured rules to the acceptance of input claim requests from remote, federated and corporate network users and the generation and issuance of a token that provides authorized access to protect resources.

3.4 Assumptions

The following assumptions provide the foundation for security objectives for the operational environment for the TOE.

Identifier	Assumption statement
A.COMMS	It is assumed that any connection between an untrusted network and the underlying servers for the federation service is appropriately secured by a firewall.
A.INSTALL	It is assumed that the TOE will be delivered, installed, configured and set up in accordance with documented delivery and installation/setup procedures.
A.ACCESS	It is assumed that the underlying server operating systems will provide access control mechanisms to restrict modification to TOE executables, the platform itself, configuration files and databases only to the administrators authorized to perform these functions.
A.I&A	It is assumed that underlying server operating systems will provide the capability to enforce identification and authentication of local administrators.
A.ATTRIBUTE	It is assumed that the IT environment will provide secure methods for storing managing and supplying identity related attributes for populating submitted claims as requested by the TOE.
A.CONFIG	It is assumed that the IT environment will provide secure methods for storing and managing TSF data for the TOE.
A.UNTRUSTED	It is assumed that no untrusted software is installed on the machines the TOE is installed on.
A.COMPETENT	It is assumed that there will be one or more competent administrators assigned to manage the TOE, its platform and the security of the information both of them contain.
A.NO_EVIL	It is assumed that the administrator(s) are not careless, willfully negligent, nor hostile, and will follow and abide by the instructions provided by the administration documentation.

Identifier	Assumption statement
A.PARTNERS	It is assumed that organizations that enter into a trust relationship are capable of providing the necessary IT environment and operational support to effectively manage their attribute store, federation service and underlying platforms.
A.CERTIFICATES	It is assumed that the IT environment and underlying server operating systems are capable of producing and securely managing the necessary cryptographic certificates required.
A.PROTECT	It is assumed that the TOE and its platform will be located within facilities providing controlled access to prevent unauthorized physical access.

4 Security objectives (ASE_OBJ)

4.1 Overview

The security objectives are a concise statement of the intended response to the security problem defined in Section 3. There are security objectives for the TOE to address and additional objectives that provide specific direction for the intended environment in which the TOE is to operate.

4.2 Security objectives for the TOE

Identifier	Objective statements
O.ADMIN	The TOE must provide administrators with the capability to manage the various security functions and policies of the TOE.
O.AUTHENTICATE	The TOE must provide the capability to authenticate all users prior to processing claims and generating a security token for the user.
O.AUTHENTICITY	The TOE must provide protection for issued security tokens to ensure that they cannot be forged or modified in order to gain unauthorized access to protected resources.
O.BINDING	The TOE must be capable of binding all authenticated users to provisioned security tokens.
O.DEFAULT	The TOE must provide default restrictive policies and authorization rules.
O.EXPORT	The TOE shall prevent claims data from being exported outside the scope of control of the TOE in an insecure manner resulting in potential loss of integrity and confidentiality of the stored identity information.
O.IMPORT	The TOE shall prevent claims data from being imported from outside the scope of control of the TOE in an insecure manner resulting in potential loss of integrity and confidentiality of the stored identity information.
O.INTERNAL	The TOE shall be capable of preventing unauthorized and modification of configuration data configuration data that is being transmitted between physically separate parts of the TOE.
O.PATH	The TOE must provide all remote and federated users with a trusted interface for authenticating to the TOE to ensure the confidentiality and integrity of user and TSF data.
O.RESTRICTED	The TOE must be capable of protecting all administrative functions of the TOE so that they can only be accessed by the authorized administrator.

Identifier	Objective statements
O.ROLES	The TOE must be capable of associating all authenticated user with a specific role that relates to their location and expected interactions with the TOE.
O.TOKEN	The TOE must be capable of applying administrator configured rules to the acceptance of input claim requests from remote, federated and corporate network users and the generation and issuing a security token that provides authorized access to protect resources.
O.TRUST	The TOE must be capable of protecting trust relationship data being transmitted between the claims provider and relying parties from unauthorized modification and/or access.

4.3 Security objectives for the environment

Identifier	Objective statements
OE.COMMS	The IT environment shall provide appropriate protection for components of the TOE from untrusted networks.
OE.INSTALL	The operational environment shall ensure that the TOE is delivered, installed, configured and set up in accordance with documented delivery and installation/setup procedures.
OE.ACCESS	The IT environment shall provide access control mechanisms to restrict modification to TOE executables, the platform itself, configuration files and databases only to the administrators authorized to perform these functions.
OE.I&A	The IT environment shall provide functionality for supporting and enforcing identification and authentication of local administrators.
OE.ATTRIBUTE	The IT environment shall provide secure methods for storing managing and supplying identity related attributes for populating submitted claims as requested by the TOE.
OE.CONFIG	The IT environment shall provide secure methods for storing and managing TSF data for the TOE.
OE.UNTRUSTED	The operational environment shall ensure that no untrusted software is installed on the machines the TOE is installed on.
OE.COMPETENT	The operational environment shall provide one or more competent administrators assigned to manage the TOE, its platform and the security of the information both of them contain.

Identifier	Objective statements
OE.NO_EVIL	The operational environment shall ensure that administrator(s) are not careless, willfully negligent, nor hostile, and will follow and abide by the instructions provided by the administration documentation.
OE.PARTNERS	The operational environment shall ensure that trusted partner organization’s are capable of providing the necessary IT environment and operational support to effectively manage their attribute store, federation service and underlying platforms.
OE.CERTIFICATES	The IT environment shall be capable of producing and securely managing the necessary cryptographic certificates required.
OE.PROTECT	The operational environment shall provide facilities that protect the TOE and its platform from unauthorized physical access.
OE.MANAGE	The IT environment shall provide an identity management tool supported by the TOE. The tool shall allow administrators to manage the identities of users of the TOE.
OE.CONFIDENTIALITY	The IT environment must be capable of protecting configuration data from unauthorized access.
OE.INTEGRITY	The IT environment must be capable of preventing unauthorized modification of configuration data that has been exported outside the scope of control of the TOE.

5 Security functional requirements (ASE_REQ)

5.1 Overview

This section defines the security requirements satisfied by the TOE. Each requirement has been extracted from version 3.1 (Rev 3) of the CC, part 2 providing functional requirements and part 3 providing assurance requirements.

Part 2 of the CC defines an approved set of operations that may be applied to security functional requirements. Following are the approved operations and the document conventions that are used within this ST to depict their application:

- **Assignment.** The assignment operation provides the ability to specify an identified parameter within a requirement. Assignments are depicted using bolded text and are surrounded by square brackets as follows [**assignment**].

- **Selection.** The selection operation allows the specification of one or more items from a list. Selections are depicted using bold italics text and are surrounded by square brackets as follows [***selection***].

- **Refinement.** The refinement operation allows the addition of extra detail to a requirement. Refinements are indicated using bolded text, for **additions**, and strike-through, for ~~deletions~~.

- **Iteration.** The iteration operation allows a component to be used more than once with varying operations. Iterations are depicted by placing a letter at the end of the component identifier as follows FDP_IFF.1a and FDP_IFF.1b.

The security functional requirements are expressed using the notation stated in Section 5.1 above and are identified in the table below.

Identifier	Title
User data protection (FDP)	
FDP_DAU.2	Data authentication with identity of guarantor
FDP_ETC.2	Export of user data with security attributes
FDP_IFC.1	Subset information flow control (CLAIMS)
FDP_IFF.1	Simple security attributes (CLAIMS)
FDP_ITC.1	Import of user data with security attributes

Identifier	Title
FDP_ITT.1	Basic internal transfer protection
Identification and authentication (FIA)	
FIA_ATD.1	User attribute definition
FIA_UAU.1	Timing of authentication
FIA_UID.2	User identification before any action
FIA_USB.1	User-subject binding
Security management (FMT)	
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
Protection of the TSF (FPT)	
FPT_ITT.1	Basic internal TSF data transfer protection
Trusted path/channels (FTP)	
FTP_ITC.1	Inter-TSF trusted channel
FTP_TRP.1	Trusted path

5.2 User data protection (FDP)

5.2.1 FDP_DAU.2 Data Authentication with identity of guarantor

Hierarchical to:	FDP_DAU.1 Basic Data Authentication
Dependencies:	FIA_UID.1 Timing of identification
FDP_DAU.2.1	The TSF shall provide a capability to generate evidence that can be used as a guarantee of the validity of [**security tokens**].
FDP_DAU.2.2	The TSF shall provide [**relying party**] with the ability to verify evidence of the validity of the indicated information and the identity of the user that generated the evidence.
Notes:	The federation server uses a token-signing certificate to digitally sign all security tokens that it produces. Because each security token is digitally signed by the claims provider, the relying party can verify that the security token was in fact issued by the claims provider and that it was not modified.

5.2.2 FDP_IFC.1 Subset information flow control (CLAIMS)

Hierarchical to:	No other components
Dependencies:	FDP_IFF.1 Simple security attributes
FDP_IFC.1.1	The TSF shall enforce the [**CLAIMS-SFP**] on [**a) Subjects:** **i. user,** **ii. relying party, and** **iii. claims provider.** **b) Information:** **i. incoming claim, and** **ii. security token.** **c) Operations:** **i. accept incoming claims,** **ii. authorizing claims, and** **iii. issuing outgoing claims through a security token**].
Notes:	This SFR is designed to model the AD FS central claims rule engine that accepts incoming claim requests, processes those claim requests and issues appropriate authorizations through a security token in accordance with specific rules that are applied to federated trust relationships.

5.2.3 FDP_IFF.1 Simple security attributes (CLAIMS)

Hierarchical to:	No other components.
Dependencies:	FDP_IFC.1 Subset information flow control FMT_MSA.3 Static attribute initialisation
FDP_IFF.1.1	The TSF shall enforce the [**CLAIMS-SFP**] based on the following types of subject and information security attributes: [**a) User (subject):** **i. UPN** **ii. E-mail** **iii. Common name** **b) Relying party and claims provider (subjects):** **i. Relying party identifier** **ii. Claims provider identifier** **iii. Acceptance transform rules** **iv. Issuance transform rules** **v. Issuance authorization rules** **vi. Delegation authorization rules** **vii. Impersonation authorization rule** **viii. Token-signing certificate** **ix. Token-decrypting certificate** **c) Incoming claims:** **i. User** **ii. Claim type** **iii. Requested access rights** **iv. Resource** **d) Security token:** **i. Issuer** **ii. Authorization claims (ClaimType, Right, Resource)** **iii. Claims provider digital signature**].
FDP_IFF.1.2	The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold: [**a) The users input claims must be accepted by the claims provider in accordance with the acceptance transform rule that is associated with the claims provider trust.** **b) If the user is requesting authorization to access a resource, then the relying party must have an issuance authorization rule that permits**

	the user to access the requested resource in order for a returned security token to include this claim. **c) If the user is requesting to impersonate another user without having to identify the requesting user in the token, then a relying party trust must have an impersonation authorization rule that permits this specific authorization in order for a returned security token to include this claim.** **d) If the user is requesting to impersonate another user while still identifying the requesting user in the token, then a relying party trust must have a delegation authorization rule that permits this specific authorization in order for a returned security token to include this claim.** **e) The relying party must permit the issuance of a security token to the user in accordance with the issuance transform rules**].
FDP_IFF.1.3	The TSF shall enforce the [**following additional rules:** **a) All claim rules are executed chronologically as they appear in a rule set. The claim rule that is at the top of the rule set is processed first and then subsequent rules are processed until all of the rules have been run.** **b) Each rule within a rule set is only executed once.** **c) All claims are processed using the following claims pipeline:** **i. Claims that are received from the claims provider are processed by the acceptance transform rules on the claims provider trust. These rules determine which claims are accepted from the claims provider.** **ii. Output from the acceptance transform rules is used as input to the issuance authorization rules. These rules determine whether the user is permitted to access the relying party.** iii. **Output from the acceptance transform rules is used as input to the issuance transform rules. These rules determine the claims that will be sent to the relying party**].
FDP_IFF.1.4	The TSF shall explicitly authorise an information flow based on the following rules: [**none**].
FDP_IFF.1.5	The TSF shall explicitly deny an information flow based on the following rules: [**a) If the user submitting the claim does not have an account in the attribute store of either the relying party or the claims provider, or** **b) If the user is unable to submit the required authentication data**].
Notes:	This SFR is designed to model the functionality of the claims pipeline process: accepting incoming claims, authorizing the claims requester and issuing outgoing claims.

5.2.4 FDP_ETC.2 Export of user data with security attributes

Hierarchical to:	No other components.
Dependencies:	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]
FDP_ETC.2.1	The TSF shall enforce the [**CLAIMS-SFP**] when exporting user data, controlled under the SFP(s), outside of the TOE.
FDP_ETC.2.2	The TSF shall export the user data with the user data's associated security attributes.
FDP_ETC.2.3	The TSF shall ensure that the security attributes, when exported outside the TOE, are unambiguously associated with the exported user data.
FDP_ETC.2.4	The TSF shall enforce the following rules when user data is exported from the TOE: [**a) The TOE is only permitted to transmit incoming claims to approved attribute stores.** **b) The TOE communicates with attribute stores in accordance with established attribute store priority.** **c) All TOE communications with approved account stores are protected by SSL/TLS**].
Notes:	This SFR provides the control associated with taking the incoming claims from the federation server and having that data transmitted to the attribute store (AD DS in most cases) for processing. AD FS is tightly integrated with AD DS. AD FS retrieves user attributes and authenticates users against AD DS. AD FS also uses Windows Integrated Authentication and security tokens that AD DS creates.

5.2.5 FDP_ITC.1 Import of user data with security attributes

Hierarchical to:	No other components.
Dependencies:	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FTP_ITC.1 Inter-TSF trusted channel, or FTP_TRP.1 Trusted path] FPT_TDC.1 Inter-TSF basic TSF data consistency
FDP_ITC.1.1	The TSF shall enforce the [**CLAIMS-SFP**] when importing user data, controlled under the SFP, from outside of the TOE.
FDP_ITC.1.2	The TSF shall ignore any security attributes associated with the user data when imported from outside the TOE.
FDP_ITC.1.3	The TSF shall enforce the following rules when importing user data controlled

	under the SFP from outside the TOE: [a) **The TOE is only permitted to import attributes from approved attribute stores.** b) **The TOE communicates with account stores in accordance with established attribute store priority.** c) **All TOE communications with approved attribute stores are protected by SSL/TLS**].
Notes:	This SFR provides the rules associated with receiving information back from the attribute store in a secure manner to populate claims.

5.2.6 FDP_ITT.1 Basic internal transfer protection

Hierarchical to:	No other components.
Dependencies:	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]
FDP_ITT.1.1	The TSF shall enforce the [**CLAIMS-SFP**] to prevent the [***disclosure and modification***] of user data when it is transmitted between physically-separated parts of the TOE.
Notes:	This SFR aims to provide protection of user data (claims data) between separate parts of the TOE (AD FS Web Agent, Federation Service Proxy and the Federation Service).

5.3 Identification and authentication (FIA)

5.3.1 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 **requesting claims**: [**a) UPN** **b) Email address** **c) Common name**].
Notes:	Users requesting access to resources will be required to provide at least the common security attributes.

5.3.2 FIA_UAU.1 Timing of authentication

Hierarchical to:	No other components.
Dependencies:	FIA_UID.1 Timing of identification.
FIA_UAU.1.1	The TSF shall allow [**presentation of claims through a security token or authentication data**] on behalf the user to be performed before the user is authenticated.
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.
Notes:	The user is free to present claims through a security token, or authentication data to claims-aware applications prior to authentication being completed. AD FS does not authenticate the user, all authentication activities are passed on to the attribute store (AD DS in most cases) to be authenticated prior to generating claims and providing authorized access to protected resources.

5.3.3 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.
Notes:	AD FS must have an indication of the identity of the user prior to being capable of undertaking any controlled actions on that user’s behalf.

5.3.4 FIA_USB.1 User-subject binding

Hierarchical to:	No other components
Dependencies:	FIA_ATD.1 User attribute definition
FIA_USB.1.1	The TSF shall associate the following user security attributes with subjects acting on the behalf of that user: [**security token and associated claims**].
FIA_USB.1.2	The TSF shall enforce the following rules on the initial association of user security attributes with subjects acting on the behalf of users: [**a) the user must be authenticated by one of the approved attribute stores** **b) the authorization rules that pertain to the relevant relying party trust must provide requested access, and** **c) the issuance rules must permit the issuance of a security token**].
FIA_USB.1.3	The TSF shall enforce the following rules governing changes to the user security attributes associated with subjects acting on the behalf of users: [**none**].
Notes:	This SFR provides the rules for associating users (corporate network, remote or federated) with the security token that represents their respective claims.

5.4 Security management (FMT)

5.4.1 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 [**CLAIMS-SFP**] to provide [***restrictive***] default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2	The TSF shall allow the [**administrator**] to specify alternative initial values to override the default values when an object or information is created.
Notes:	This SFR ensures that the default position relating to both access control to resources and the control of issuing tokens is one of deny.

5.4.2 FMT_MTD.1 Management of TSF data

Hierarchical to:	No other components
Dependencies:	FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions
FMT_MTD.1.1	The TSF shall restrict the ability to [***query, modify delete,*** **[create]**] the [**following TSF data:** **a) attribute store configuration data,** **b) trust policy,** **c) claims ruleset,** **d) Federation Server configuration data,** **e) Federation Server Proxy configuration data, and** **f) certificates**] to [**the administrator**].
Notes:	The administrator is the only role permitted to interact with the TOE to perform general security, administrative and operational activities.

5.4.3 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 management functions: [

	a) **Relying party trust management (ADFSRelyingPartyTrust)** b) **Claims provider trust management (ADFSClaimsProviderTrust)** c) **Attribute store management (ADFSAttributeStore)** d) **Claim description management (ADFSClaimDescription)** e) **Endpoint management (ADFSEndpoint)** f) **Certificate management (ADFSCertificate)** g) **Federation proxy configuration management (ADFSProxyProperties)** h) **Federate server configuration management (ADFSProperties)** i) **Claim ruleset management (ADFSClaimRuleSet)**].
Notes:	The TOE provides a graphical user interface (GUI) tool—the AD FS 2.0 Management snap-in for Microsoft Management Console (MMC). Windows PowerShell™ cmdlets that are also included with AD FS 2.0 as an optional toolset to configure and administer the TOE.

5.4.4 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 [a) **corporate network user,** b) **remote user,** c) **federated user, and** d) **administrator**].
FMT_SMR.1.2	The TSF shall be able to associate users with roles.
Notes:	In the context of users (for example employees) that are able to access the functionality of the TOE there are several different types. Individuals that are logged on to an Active Directory forest in the corporate network are considered to be corporate network users. Remote users are also employees who are logged on to an Active Directory domain, however, they are accessing this domain remotely, that is from outside the corporate network environment. A federated user is generally not an employee and their account will reside in an account partner organization, who can access federated applications that reside in a resource partner organization. The administrator relates simply to the local administrator that has privileged access to the TOE and the underlying and supporting platforms.

5.5 Protection of the TSF (FPT)

5.5.1 FPT_ITT.1 Basic internal TSF data transfer protection

Hierarchical to:	No other components.
Dependencies:	No dependencies.
FPT_ITT.1.1	The TSF shall protect TSF data from [***disclosure and modification***] when it is transmitted between separate parts of the TOE.
Notes:	This SFR ensures that the AD FS solution is capable of federation service metadata (TSF data) when transferred between components of the solution that may be implemented on physically separate servers, namely the Federation Service Proxy and the Federation Service.

5.6 Trusted path/channels (FTP)

5.6.1 FTP_ITC.1 Inter-TSF trusted channel

Hierarchical to:	No other components
Dependencies:	No dependencies
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 [***the TSF or 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 [**communicating trust relationship data**].
Notes:	This SFR implements a method for securely communicating between the claims provider and the relying party in any trust relationship.

5.6.2 FTP_TRP.1 Trusted path

Hierarchical to:	No other components.
Dependencies:	No dependencies.
FTP_TRP.1.1	The TSF shall provide a communication path between itself and [***remote***] users that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from [***modification and disclosure***].

FTP_TRP.1.2	The TSF shall permit [***remote users***] to initiate communication via the trusted path.
FTP_TRP.1.3	The TSF shall require the use of the trusted path for [***all users presenting a security token or authentication data to a claims-aware application***].
Notes:	This SFR implements security functionality to ensure that the user (either federated or remote) is capable of establishing a trusted path between itself in a remote location via a browser and the claims-aware application. This is through the implementation of an SSL/TLS session to protect communications between the user and the application from modification and disclosure. Each AD FS-enabled Web server that hosts an AD FS Web Agent uses SSL server authentication certificates to securely communicate with Web clients.

6 Security assurance requirements (ASE_REQ)

This ST implements the Security Assurance Requirements (SARs) of the EAL4 package and augments this package with the inclusion of the ALC_FLR.3, systematic flaw remediation.

EAL4 provides assurance by a full security target and an analysis of the SFRs in that ST, using a functional and complete interface specification, guidance documentation and a description of the modular design of the TOE. The full implementation is also provided to the evaluator so that analysis can be conducted of an evaluator-selected subset, so that security behaviour can be understood and potential vulnerabilities identified.

The analysis is supported by independent testing of the TSF, which can be based on evidence of developer testing of the functions of the TOE. In addition, the evaluators will conduct a vulnerability analysis using all provided inputs and ensure that the TOE is resistant to penetration attackers with an ***enhanced-basic*** attack potential. EAL4 also provides assurance through the use of development environment controls and additional TOE configuration management including automation, and evidence of secure delivery procedures.

The selected set of SARs is appropriate due to the intended enterprise operating environment and customer base that this product is intended for. EAL4 provides evaluators with access to the implementation details for the TOE and enables deep analysis to identify potential vulnerabilities and exposures which is relevant and expected of an enterprise-grade software product.

EAL4 provides the right balance with understanding and documenting the modular structure of the TOE and the implementation detail, and providing sufficient assurance through independent functional and penetration testing. The following table highlights the assurance requirements of the EAL4 assurance package.

Assurance class	Assurance components
ADV: Development	ADV_ARC.1 Security architecture description
	ADV_FSP.4 Complete functional specification
	ADV_IMP.1 Implementation representation of the TSF
	ADV_TDS.3 Basic modular design
AGD: Guidance documents	AGD_OPE.1 Operational user guidance
	AGD_PRE.1 Preparative procedures
ALC: Life cycle support	ALC_CMC.4 Production support, acceptance procedures and automation

Assurance class	Assurance components
	ALC_CMS.4 Problem tracking CM coverage
	ALC_DEL.1 Delivery procedures
	ALC_DVS.1 Identification of security measures
	ALC_LCD.1 Developer defined life-cycle model
	ALC_TAT.1 Well-defined development tools
	ALC_FLR.3 Systematic flaw remediation
ASE: Security Target evaluation	ASE_CCL.1 Conformance claims
	ASE_ECD.1 Extended components definition
	ASE_INT.1 ST Introduction
	ASE_OBJ.2 Security objectives
	ASE_REQ.2 Derived security requirements
	ASE_SPD.1 Security Problem Definition
	ASE_TSS.1 TOE summary specification
ATE: Tests	ATE_COV.2 Analysis of coverage
	ATE_DPT.1 Testing: basic design
	ATE_FUN.1 Functional testing
	ATE_IND.2 Independent testing – sample
AVA: Vulnerability assessment	AVA_VAN.3 Focused vulnerability analysis

7 TOE summary specification

7.1 Overview

This chapter provides the TOE summary specification, a high-level description of how the TOE implements the claimed security functional requirements. The TOE implements the following security functions that suitably address the claimed set of requirements:

a) **Claims policy management.** AD FS provides a claims rules engine that is used to manage claims policy for the implementation of the capability. In the context of digital identities, claims are statements that one subject makes about itself or another subject. These claims can be made by a person directly or provided to others by a third party. Other parties can rely on the values of the claims to perform a process of digital identification.

b) **Trust management.** AD FS implements the capability to manage cross-organizational (federation-based) collaboration. The determination of the trust relationship depends on whether the organisation will host a web resource to be accessed by other organizations across the Internet—or the reverse.

c) **Token issuance.** AD FS implements a security token service (STS) that processes all claims and requests for tokens.

7.2 Claims policy management

In the context of digital identities, claims are statements that one subject (a person, organization, or thing) makes about itself or another subject. For example, claims can be made about the name, age, role, or other characteristics of a person. These claims can be made by a person directly or provided to others by a third party. Other parties can rely on the values of the claims to perform a process of digital identification.

At its most basic level, AD FS works with claims and uses its Federation Service in the following ways:

a) When a Federation Service is configured in the claims provider role, it serves as a claims producer—authenticating users and issuing outgoing claims on their behalf. In this role, the Federation Service can retrieve claims data from an attribute store and then send that information back in the form of tokens.

b) When a Federation Service is configured in the relying party role, it can also serve as a claims consumer—processing and trusting the incoming tokens that other claims providers pass to it. While relying parties can often simply be applications that are claims aware and that are able to process these tokens, in this role, AD FS 2.0 also supports federated identity scenarios in which a relying party validates or handles claims that another claims provider issues. More precisely, a Federation Service in the relying party role looks at and validates claims that some other Federation Service asserts and, upon successful validation, it either reaffirms those claims to its relying parties or it asserts additional or even different claims in the token that it issues.

The TOE provides an MMC snap-in to administer AD FS. There is also a suite of cmdlets that can be run through Windows PowerShell. These resources can be used to configure or administer a federation server or federation server proxy. A resource is implemented as an object type that is used to derive one or more cmdlets.

7.3 Trust management

Adding a claims provider trust to AD FS gives users of that claims provider access to the relying parties that are configured in AD FS. Each relying party application makes authorization decisions about a user by examining the claims that AD FS provides. AD FS uses the administrator-defined claim rules for a claims provider to determine what claims to issue about each user, based on the relying party that is involved.

In AD FS, a relying party is a Federation Service or application that consumes claims in a particular transaction. Claims that originate from a claims provider can be presented and consumed by the relying party.

A Federation Service or application in a relying party role:

a) Acts as a Web service that can request a set of claims from a trusted claims provider.

b) Consumes the claims that it receives from its configured claims provider.

When AD FS is configured in the role of the relying party, it acts as a partner that trusts a claims provider to authenticate users. Therefore, the relying party consumes the claims that are packaged in security tokens that come from users in the claims provider.

Typically, the Federation Service in the relying party role uses the security tokens that the claims provider produces to issue tokens to the Web servers that are located in the same organization.

To function as a relying party application for AD FS, the relying party Web server must have either the Windows Identity Foundation (WIF) platform installed or the AD FS 1.0/1.1 claims-aware Web agent. Web servers that function as a relying party application can host claims-aware applications.

7.4 Token issuance

In AD FS, a claims provider is a Federation Service responsible for collecting and authenticating a user, building up claims for that user, and packaging the claims into security tokens that the relying party uses to make authorization decisions.

A Federation Service in a claims provider role provides the following:

a) a web service that issues security tokens in a recognized format, and

b) administrators with the means to publish federation metadata that a relying party can retrieve.

The claims pipeline in AD FS represents the path that claims must follow through the Federation Service before they can be issued. The Federation Service manages the entire end-to-end process of flowing claims through the various stages of the claims pipeline, which also includes the processing of claim rules by the claim rule engine.

The claims pipeline process consists of three high-level stages. Each stage in this process initializes the claim rule engine to process claim rules that are specific to that stage. These stages include (in the order that they occur):

a) **Accepting incoming claims.** This stage in the claims pipeline is used to extract the incoming claims from the token and eliminate claims that are not expected or trusted. After they are extracted, the acceptance rules that make up the acceptance transform rule set for a claims provider trust are run. These rules can be used to pass through or add new claims that can then be used in the subsequent stages of the claims pipeline. The output of this stage is used as an input to second and third stage.

b) **Authorizing the claims requester**. This stage is used by the claims engine to issue permit or deny claims based on whether the token requester is allowed to obtain a token for the given relying party or not. However, before this can occur the authorization rules that make up either the issuance authorization rule set or the delegation authorization rule set for a relying party trust are ran.

c) **Issuing outgoing claims.** This stage is used to issue outgoing claims and send them along the pipeline where they will be packaged into a security token. However, before this can occur the issuance rules that make up the issuance transform rule set for a relying party trust are ran, which will determine what claims will be issued as outgoing claims.

All three stages above perform claims rules processing but use a different set of rules. As described above, each stage has an associated set of rules based on either the issuer of the incoming claims (the acceptance rules) or the target service for which the claims/token are being issued (authorization and issuance rules).

Annex A - Defined terms (ASE_REQ)

The following table defines all subjects, objects, operations, security attributes, external entities and other key terms that are used within the statements of security functional and assurance requirements.

Term/Acronym	Definition
acceptance transform rules	The set of claim rules that correspond to a particular claims provider trust. These rules define what claims from the claims provider will be accepted and used later by the issuance transform rules.
AD FS configuration database	A database that stores all the configuration data that represents a single instance of AD FS 2.0 (the Federation Service). This configuration data can be stored either in the Windows Internal Database, which is included with Windows Server 2008 and Windows Server 2008 R2, or in a Microsoft SQL Server database.
account store	AD FS 2.0 uses account stores to log on users and extract security claims for those users. Multiple account stores can be configured for a single Federation Service. You can also define their priority. The Federation Service uses Lightweight Directory Access Protocol (LDAP) to communicate with account stores. AD FS supports the following two account stores: • Active Directory Domain Services (AD DS) • Active Directory Lightweight Directory Services (AD LDS)
attribute store	A database or directory service that contains attributes about clients. These attributes can be used to issue claims about the clients. For example, AD FS 2.0 supports the use of either AD DS or SQL Server as the attribute store for a claims provider.
claim	A statement that one subject makes about itself or another subject. For example, the statement can be about a name, identity, key, group, privilege, or capability. Claims have a provider that issues them, and they are given one or more values. They are also defined by a claim value type and, possibly, associated metadata.
claim descriptions	The list of claims that AD FS 2.0 maintains for the sake of publishing federation metadata, issuing display tokens, and assisting in the authoring of claim rules.
claim issuer	The claims provider that issued the claim.
claim name	A user-friendly name for the claim type.

Term/Acronym	Definition
claim rule	A rule that is created with a claim rule template or that is written using the claim rule language in AD FS 2.0 that defines how to generate, transform, pass through, or filter claims.
claim rule template	A template that is designed to help administrators easily select and create the most appropriate claim rules for a particular business need. Claim rule templates are used only during the claim rule creation process.
claim rule language	The language that AD FS 2.0 uses to author and process the logic in all claim rules.
claim rule set	A grouping of one or more claim rules for a given federated trust that defines how claims will be processed by the claims rule engine.
claim type	The type of statement in the claim that is made. Example claim types include FirstName and Role. The claim type provides context for the claim value, and it is usually expressed as a Uniform Resource Identifier (URI).
claim value	The value of the statement in the claim that is made. For example, if the claim type is Role, a value might be Contributor.
claim value type	The type of value in the claim. For example, if the claim value is Contributor, the claim type value is String.
claims-aware application	A relying party software application that uses claims to manage identity and access for users.
claims provider	A Federation Service that issues claims for a particular transaction.
claims provider trust	In the AD FS 2.0 snap-in, a claims provider trust is a trust object that is created to maintain the relationship with another Federation Service that provides claims to this Federation Service.
client	The user—or the software of a user—that acts on claims that it receives from the claims provider.
custom attribute store	A Microsoft .NET Framework assembly component that was developed for extending the functionality of AD FS 2.0 attribute stores.
custom rule	A claim rule that you author using the claim rule language to express a series of complex logic conditions. You can build custom rules by typing the claim rule language syntax in the Send Claims Using a Custom Rule template.
delegation authorization rules	The set of claim transformation rules corresponding to a relying party trust that determines whether the requester is permitted to impersonate a user while still identifying the requester to the relying party.
digital identity	A set of claims that represent a subject.

Term/Acronym	Definition
endpoint	Endpoints provide access to the federation server functionality of Active Directory Federation Services (AD FS) 2.0, such as token issuance, and the publishing of federation metadata.
federation metadata	The data format for communicating configuration information between a claims provider and a relying party to facilitate automated configuration of claims provider trusts and relying party trusts. The data format is defined in Security Assertion Markup Language (SAML) 2.0, and it is extended in WS-Federation.
federation server	A computer running Windows Server 2008 or Windows Server 2008 R2 that has been configured using the AD FS 2.0 Federation Server Configuration Wizard to act in the federation server role. A federation server issues tokens and serves as part of a Federation Service.
federation server proxy	A computer running Windows Server 2008 or Windows Server 2008 R2 that has been configured with the AD FS 2.0 Federation Server Proxy Configuration Wizard to act as an intermediary proxy service between an Internet client and a Federation Service that is located behind a firewall on a corporate network.
Federation Service	A logical instance of AD FS 2.0. A Federation Service can be deployed as a stand-alone federation server or as a load-balanced federation server farm.
identifier	A Uniform Resource Identifier (URI) that is used to identify an object. The object can be the instance of AD FS 2.0, a claims provider, or a relying party.
identity delegation	A feature in AD FS 2.0 that makes it possible for a user or computer to be authorized to act as another user or computer to a relying party.
impersonation authorization rules	The set of claim rules corresponding to a relying party trust that determines whether the requester is permitted to impersonate a user without identifying the requester to the relying party. These rules can be created only using the Windows PowerShell™ command-line interface.
input claim set	A collection of claims within the context of a given claim rule set that is available as input to subsequent claim rules within that set. Claims in this collection are discarded after the rules are processed. The rules processing engine adds the claims that each rule generates to the input claim set so that subsequent rules within a given rule set can use those claims.
issuance rules	The set of rules applied when outgoing claims are issued across all federated trust relationships.
issuance authorization rules	The set of claim rules corresponding to a relying party trust that determines whether the requester is permitted to receive a token.
issuance transform rules	The set of claim rules that correspond to a relying party trust that determine the claims that are issued to the relying party.

Term/Acronym	Definition
output claim set	A collection of claims within the context of a given claim rule set that will determine which claims are emitted from the list of claim rules within a rule set. If temporary claims are needed for processing, a rule can be authored in such a way that the resulting claims are added to the input claim set only.
primary federation server	A computer running Windows Server 2008 or Windows Server 2008 R2 that has been configured in the federation server role with the AD FS 2.0 Federation Server Configuration Wizard and that has a read/write copy of the AD FS configuration database. You create the primary federation server when you use the AD FS 2.0 Federation Server Configuration Wizard, select the option to create a new Federation Service, and make that computer the first federation server in a federation server farm. All other federation servers in the farm must replicate changes that are made on the primary federation server to a read-only copy of the AD FS configuration database that they store locally. The term "primary federation server" does not apply when the AD FS configuration database is stored in a SQL Server database, because all federation servers can read and write equally to the SQL Server database.
relying party	A Federation Service or application that consumes claims in a particular transaction.
relying party application	Software that can consume claims to make authentication and authorization decisions. The relying party application receives the claims from a claims provider.
relying party trust	In the AD FS 2.0 snap-in, a relying party trust is a trust object that is created to maintain the relationship with a Federation Service or application that consumes claims from this Federation Service.
rich client	A client that can use the WS-Trust protocol.
Security Assertion Markup Language (SAML) Security Token	The data format for communicating claims between a claims provider and a relying party. AD FS 2.0 uses both SAML 1.1 and SAML 2.0 formats.
Security Assertion Markup Language (SAML)	The WebSSO protocol that is defined in the SAML 2.0 Core specification. The SAML protocol specifies how to use HTTP Web browser redirects to exchange assertions data. SAML is used to authenticate and authorize users across secure boundaries.
subject	A person, organization, or thing that is described or dealt with.
trust establishment	A process by which trust relationships are established between claims providers, such as AD FS 2.0, and relying party applications. This process involves the exchange of identifying certificates that make it possible for the relying party to trust the contents of claims that the claims provider issues.

Term/Acronym	Definition
trust monitoring	A feature in AD FS 2.0 that keeps the configuration of a claims provider or relying party up to date by periodically monitoring its Federation Metadata.
trust policy	The Active Directory Federation Services (ADFS) trust policy file defines the set of parameters that a Federation Service requires to identify partners, certificates, account stores, claims, and various properties of these entities that are associated with the Federation Service.
trust relationship	A federation trust relationship is the embodiment of a business-level agreement or partnership between two organizations.
User Principal Name (UPN)	An identifier used by Microsoft Active Directory that provides a user name and the Internet domain with which that username is associated in an e-mail address format. The format is [*AD username*]@[associated *domain*]; an example would be *john.smith@microsoft.com*.
Uniform Resource Locator (URL)	The address that is used to locate a Web site. URLs are text strings that must conform to the guidelines in RFC 2396.
Web browser client	A client that can use the SAML WebSSO protocol and the WS-Federation passive protocol. Also referred to as a "passive client."
Web Service Description Language (WSDL)	The data format for specifying how a Simple Object Access Protocol (SOAP) service should be called. AD FS 2.0 uses WSDL 1.1.
Windows Communication Foundation (WCF)	The Microsoft unified programming model for building service-oriented applications. Developers can use WCF to build secure, reliable, transacted solutions that integrate across platforms and interoperate with existing programs.
Windows Identity Foundation (WIF)	A framework for building identity-aware applications. The framework abstracts the WS-Trust and WS-Federation protocols and presents developers with application programming interfaces (APIs) for building security token services (STSs) and claims-aware applications. Applications can use WIF to process tokens that are issued from STSs and make identity-based decisions at the Web application or Web service.
WS-Federation	The OASIS standard specification that defines the WS-Federation Passive protocol and other protocol extensions that are used for federation.
WS-Federation Passive	The protocol for requesting claims from a claims provider by using HTTP Web browser redirects. This protocol is described in section 13 of the WS-Federation 1.2 specification.
WS-SecurityPolicy	An XML-based specification that describes the security requirements of a Web service. These security requirements include descriptions of the claims that the service requires.

Term/Acronym	Definition
WS-Trust	The SOAP protocol, which is defined by the WS-Trust specifications, for requesting claims from a claims provider. AD FS 2.0 uses both the February 2005 and 1.3 versions of the protocol.

Annex B - Correspondence and rationale

B.1 TOE security objectives rationale

The following table demonstrates that all security objectives for the TOE trace back to the threats and OSPs in the security problem definition.

Threats/OSPs	Objectives	Rationale
P.ADMIN	O.ADMIN	Provides direct mapping to the policy statement and aims to ensure that the TOE provides administrators with the capability to manage the various security functions and policies of the TOE.
P.AUTHENTICATE	O.AUTHENTICATE	Provides direct mapping to the policy statement and aims to ensure that users requesting claims are effectively authenticated by the IT environment.
P.BINDING	O.BINDING	Provides direct mapping to the policy statement and aims to ensure that users are bound to their security tokens.
P.DEFAULT	O.DEFAULT	Provides direct mapping to the policy statement and aims to ensure that the TOE provides a default policy rules of deny all for all created identity resource objects.
P.PATH	O.PATH	Provides direct mapping to the policy statement and aims to ensure that the TOE provides all remote users with a trusted interface for authenticating to the TOE to ensure the confidentiality and integrity of user and TSF data.
P.RESTRICTED	O.RESTRICTED	Provides direct mapping to the policy statement and aims to ensure that access to TSF data is restricted to the administrator.
P.ROLES	O.ROLES	Provides direct mapping to the policy statement and aims to ensure the TOE is capable of binding users to specific roles.
P.TOKEN	O.TOKEN	Provides direct mapping to the policy statement and aims to ensure there are specific rules about the acceptance, authorization of claims and the issuance of a security token.

Threats/OSPs	Objectives	Rationale
T.AUTHENTICITY	O.AUTHENTICITY	Provides direct mapping to the threat and aims to prevent an unauthorized individual from forging a security token.
T.CONFIDENTIALITY	OE.CONFIDENTIALITY	Provides direct mapping to the threat and aims to ensure that configuration data transferring between the TOE and the configuration database is well protected against eavesdropping.
T.EXPORT	O.EXPORT	Provides direct mapping to the threat and aims to prevent claims data from being exported outside the scope of control of the TOE in an insecure manner.
T.IMPORT	O.IMPORT	Provides direct mapping to the threat and aims to prevent claims information from being imported from outside the scope of control of the TOE in an insecure manner.
T.INTEGRITY	OE.INTEGRITY	Provides direct mapping to the threat and aims to ensure that configuration data transferring between the TOE and the configuration database is well protected against unauthorized modification.
T.INTERNAL	O.INTERNAL	Provides direct mapping to the threat and aims to prevent user and TSF data from being compromised when being transmitted between physically separate servers.
T.TRUST	O.TRUST	Provides direct mapping to the threat and aims to ensure that the end-user has a trusted method for engaging with the TOE.

B.2 Environment security objectives rationale

The following table demonstrates that all security objectives for the operational environment all trace back to assumptions or OSPs in the security problem definition.

Assumptions	Objectives	Rationale
A.COMMS	OE.COMMS	This IT environment objective provides a direct mapping to the stated assumption and aims to ensure that the communication channels between all server roles are appropriately secured.
A.INSTALL	OE.INSTALL	This operational environment objective provides a direct mapping to the stated assumption and aims to ensure that the TOE is delivered, installed, configured and set up in accordance with documented delivery and installation/setup procedures.
A.ACCESS	OE.ACCESS	This IT environment objective provides a direct mapping to the stated assumption and aims to ensure that the underlying server operating systems will provide access control mechanisms to restrict modification to TOE executables, the platform itself, configuration files and databases only to the administrators authorized to perform these functions.
A.I&A	OE.I&A	This IT environment objective provides a direct mapping to the stated assumption and aims to ensure that the underlying server platform provides the necessary identification and authentication capabilities.
A.ATTRIBUTE	OE.ATTRIBUTE	This IT environment objective provides a direct mapping to the stated assumption and aims to ensure that secure methods for storing managing and supplying identity related attributes for populating submitted claims as requested by the TOE are provided.
A.CONFIG	OE.CONFIG	This IT environment objective provides a direct mapping to the stated assumption and aims to ensure that secure methods for storing and managing TSF data for the TOE.
A.UNTRUSTED	OE.UNTRUSTED	This operational environment objective provides a direct mapping to the stated assumption and aims to ensure that no untrusted software is installed on the machines the TOE is installed on.
A.COMPETENT	OE.COMPETENT	This operational environment objectives provides a direct mapping to the stated assumption and aims to ensure that the administrators are competent and follow the TOE guidance.

Assumptions	Objectives	Rationale
A.NO_EVIL	OE.NO_EVIL	This operational environment objective provides a direct mapping to the stated assumption and aims to ensure that there are suitable administrator resources available to manage the TOE.
A.PARTNERS	OE.PARTNERS	This operational environment objective provides a direct mapping to the stated assumption and aims to ensure that trusted partner organizations are capable of providing the necessary IT environment and operational support to effectively manage their attribute store, federation service and underlying platforms.
A.CERTIFICATES	OE.CERTIFICATES	This IT environment objective provides a direct mapping to the stated assumption and aims to ensure that it is capable of producing and securely managing the necessary cryptographic certificates required.
A.PROTECT	OE.PROTECT	This operational environment objective provides a direct mapping to the stated assumption and aims to ensure that the TOE and its underlying platform are located within facilities providing controlled access to prevent unauthorized physical access.
P.MANAGE	OE.MANAGE	Provides direct mapping to the policy statement and aims to ensure that the IT environment provides administrators with the capability to perform the necessary identity management related tasks.

B.3 Security functional requirements rationale

The following table demonstrates that all security functional requirements trace back to security objectives of the TOE as specified in the security problem definition.

Objectives	SFRs	Rationale
O.ADMIN	FMT_SMF.1	This SFR specifies the suite of security functions and policy management capabilities that must exist within the TOE for the Administrator.
O.AUTHENTICATE	FIA_UAU.1	This SFR provides the specification relating to authenticating users prior to processing claims and generating a security token for the user.
	FIA_UID.2	This SFR provides the required identification function that precedes the authentication function.
O.AUTHENTICITY	FDP_DAU.2	This SFR provides protection for issued security tokens to ensure that they cannot be forged or modified in order to gain unauthorized access to protected resources.
O.BINDING	FIA_ATD.1	This SFR specifies the attributes that are associated with users to support the binding of the token to the user.
	FIA_USB.1	This SFR specifies the rules that apply when generating and binding a token to a user.
O.DEFAULT	FMT_MSA.3	This SFR specifies the requirements associated with establishing default restrictive access and federation requirements from initialization.
O.EXPORT	FDP_ETC.2	This SFR specifies the rules around controlling claims data (user data) when it is transferred outside the scope of control of the TOE.
O.IMPORT	FDP_ITC.1	This SFR specifies the rules around controlling claims data (user data) when it is imported from outside the scope of control of the TOE.
O.INTERNAL	FDP_ITT.1	This SFR specifies controls for the protection of user data (claims and token data) internally as it flows between physically separate components of the TOE.

Objectives	SFRs	Rationale
	FPT_ITT.1	This SFR specifies controls for the protection of TSF data (trust-related data) internally as it flows between physically separate components of the TOE.
O.PATH	FTP_TRP.1	This SFR specifies the implementation of a trusted path between the end-user and the TOE.
O.RESTRICTED	FMT_MTD.1	This SFR specifies that controls needed to limit the security administration and management functions to the administrator.
O.ROLES	FMT_SMR.1	This SFR simply identifies the key roles associated with the TOE, the user and the administrator.
O.TOKEN	FDP_IFC.1	This SFR specifies the set of information, subjects and security attributes that relate to the implementation of an information flow control policy for the receipt, generation and issuance of tokens.
	FDP_IFF.1	This SFR specifies the actual information flow control rules associated with the claims policy of the TOE.
O.TRUST	FTP_ITC.1	This SFR specifies controls for protecting trust information that flows between the federation servers.

B.4 Dependency rationale

SFR	Dependencies	Rationale
FDP_DAU.2	FIA_UID.1 Timing of identification	FIA_UID.2 included
FDP_ETC.2	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]	FCP_IFC.1 included
FDP_IFC.1	FDP_IFF.1 Simple security attributes	Included
FDP_IFF.1	FDP_IFC.1 Subset information flow control FMT_MSA.3 Static attribute initialisation	Included Included
FDP_ITC.1	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] [FTP_ITC.1 Inter-TSF trusted channel, or FTP_TRP.1 Trusted path] FPT_TDC.1 Inter-TSF basic TSF data consistency	Included Included Included Not included. This requirements deals with the protection of user data in the form of attributes that are applied to security tokens.
FDP_ITT.1	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]	Included Included
FIA_ATD.1	No dependencies	-
FIA_UAU.1	FIA_UID.1 Timing of identification	FIA_UID.2 included
FIA_UID.2	No dependencies	-
FIA_USB.1	FIA_ATD.1 User attribute definition	Included
FMT_MSA.1	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions	Included Included Included Included
FMT_MSA.3	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles	Not included. Functionality and control of security attributed adequately covered through both FMT_MTD SFRs that specify the TSF data that is controlled by both the user and the administrator. Included

SFR	Dependencies	Rationale
FMT_MTD.1	FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions	Included Included
FMT_SMF.1	No dependencies	-
FMT_SMR.1	FIA_UID.1 Timing of identification	FIA_UID.2 included
FPT_ITT.1	No dependencies	-
FPT_TDC.1	No dependencies	-
FTP_ITC.1	No dependencies	-
FTP_TRP.1	No dependencies	-