Cisco Systems, Inc.
ACE XML Gateway and Manager
Version 5.0.3

Security Target

Version 1.0

07/25/08

Prepared for:

Cisco Systems, Inc.

170 West Tasman Drive
San Jose, CA 95134

Prepared By:

Cisco Systems, Inc.

and

Science Applications International Corporation

Common Criteria Testing Laboratory

7125 Columbia Gateway Drive, Suite 300
Columbia, MD 21046

LIST OF TABLES

1 Security Target Introduction

This section provides the Security Target (ST) and Target of Evaluation (TOE) identification, ST conventions, ST conformance claims, and the ST organization. The TOE is the ACE (Application Control Engine) XML Gateway and Manager Version 5.0.3 provided by Cisco Systems, Inc.

The Security Target contains the following additional sections:

- TOE Description (Section 2)
- Security Environment (Section 3)
- Security Objectives (Section 4)
- IT Security Requirements (Section 5)
- TOE Summary Specification (Section 6)
- Protection Profile Claims (Section 7)
- Rationale (Section 8)

1.1 Security Target, TOE and CC Identification

ST Title – Cisco Systems, Inc. ACE XML Gateway and Manager Version 5.0.3 Security Target

ST Version – Version 1.0

ST Date – 07/25/08

TOE Identification – Cisco Systems ACE XML Gateway and Manager Version 5.0.3

CC Identification – Common Criteria for Information Technology Security Evaluation, Version 2.3, August 2005, January 2004.

1.2 Conformance Claims

This TOE is conformant to the following CC specifications:

- Common Criteria for Information Technology Security Evaluation Part 2: Security Functional Requirements, Version 2.3, August 2005.
  - Part 2 Conformant
- Common Criteria for Information Technology Security Evaluation Part 3: Security Assurance Requirements, Version 2.3, August 2005.
  - Part 3 Conformant
  - EAL 3 augmented with ALC_FLR.2

1.3 Conventions

The following conventions have been applied in this document:

- Security Functional Requirements – Part 2 of the CC defines the approved set of operations that may be applied to functional requirements: iteration, assignment, selection, and refinement.
    - Iteration: allows a component to be used more than once with varying operations. In the ST, iteration is indicated by a letter in parenthesis placed at the end of the component. For example FDP_ACC.1a and FDP_ACC.1b indicate that the ST includes two iterations of the FDP_ACC.1 requirement, a and b.
    - Assignment: allows the specification of an identified parameter. Assignments are indicated using bold and are surrounded by brackets (e.g., [**assignment**]).
    - Selection: allows the specification of one or more elements from a list. Selections are indicated using bold italics and are surrounded by brackets (e.g., [***selection***]).
    - Refinement: allows the addition of details. Refinements are indicated using bold, for additions, and strike-through, for deletions (e.g., "… **all** objects …" or "… ~~some~~ **big** things …").
- Other sections of the ST – Other sections of the ST use bolding to highlight text of special interest, such as captions.

1.4 Terminology

This document uses the following terminology:

3DES

Also known as Triple DES, a NIST-standard cryptographic cipher that uses a 168-bit key and multiple-pass operations to provide increased security over the single-pass, 56-bit DES (Data Encryption Standard) algorithm on which it is based. The complete specification for DES and 3DES is defined by the ANSI X9.52-1998 standard. Optionally, administrators of the Web services SFP can specify that the ACE XML Gateway use 3DES to encrypt outgoing messages.

ACE XML appliance

A Hewlett-Packard DL360 G5 chassis configured at the Cisco Systems factory with the operating system, hardware cryptomodule, TOE software, firmware, and local storage required to function as an instance of the Cisco Systems ACE XML Gateway and Manager Version 5.0.3.

ACE XML Gateway appliance

A Cisco ACE XML Gateway and Manager Version 5.0.3 appliance configured as a dedicated ACE XML Gateway instance according to the instructions and specifications in the "Creating the Common Criteria Evaluated Configuration" chapter of the administrator guidance. The ACE XML Gateway appliance (Gateway) is the component of the TOE that processes message traffic according to the Web Services SFP that authorized administrators define. In the evaluated configuration, no traffic can enter the trusted network without flowing through the Gateway according to all rules the Web services SFP defines. The Gateway rejects messages that do not conform to the rules the Web services SFP defines.

ACE XML Manager appliance

A Cisco ACE XML Gateway and Manager Version 5.0.3 appliance configured as a dedicated ACE XML Manager instance according to the instructions and specifications in the "Creating the Common Criteria Evaluated Configuration" chapter of the administrator guidance. The ACE XML Manager appliance (Manager) is the component of the TOE that authorized administrators use to define a Web Services SFP and load it onto one or more ACE XML Gateway instances, which enforce the Web Services SFP.

Authenticator

A security policy component that specifies a collection of subject security attributes and values that positively identify a message sender to the Web Services SFP. An incoming message must satisfy all of the requirements of a defined authenticator as a prerequisite to further processing by a handler in the same authorization group as the authenticator. An authenticator filters messages (FDP) on values in the headers of a message incoming to the Gateway. Authenticators are not user accounts and a consumer who sends a message to a service the ACE XML Gateway protects has not logged on to the TOE.

Authorization Group

A representation of a group composed of authenticators and handlers. Authenticators in an authorization group can access a common set of handlers that route messages to protected services. Satisfying the requirements of an authenticator in the group makes an incoming message eligible for further processing by one of the handlers in the group. The message is not available to handlers outside of the authorization group. The TOE provides authorization groups to ease management of access permissions and to organize authenticators for convenience.

Client Certificate

The X.509 certificate that authenticates a client to a server; for example, the certificate that authenticates the sender of a message to the Gateway. Administrators of the Web services SFP can specify that the Gateway use the Distinguished Name value to authenticate the sender of a message for purposes of establishing an SSL connection to the Gateway for processing the message.

Cluster

A group of ACE XML Gateway instances and the ACE XML Manager instance that controls them.

Consumer

A client that connects to the ACE XML Gateway in an attempt to gain access to its protected services. Clients do not log into the TOE.

Cryptomodule

A hardware module that includes a processor specialized for generating, storing and using keys for cryptographic operations.

Decryption

The process of recovering the original text of an encrypted message.

Denial-of-service attack

An attack in which a service is flooded with so many requests that it becomes unavailable to legitimate users. To prevent Denial-of-Service attacks, the TOE monitors the frequency of incoming requests; when the rate of requests from a particular IP address exceeds a policy-specified threshold, the TOE blocks requests from that address for a policy-specified amount of time.

Distinguished Name

A structured string used as the name of the entity referred to by an X.509 certificate or an LDAP directory. DNs are intended to be unique, and are structured in a way that is designed to make it extremely unlikely that the same DN could refer to two different entities. The fields of a DN are defined by the Internet Engineering Task Force document RFC 1979, "A String Representation of Distinguished Names".

DSA

Digital Signature Algorithm, the algorithm used by the U. S. government Digital Signature Standard (DSS) to generate and verify digital signatures. The FIPS PUB 186 Federal Information Processing Standards publication defines the DSA specification.

Encryption
The process by which message text is converted to a coded equivalent, called “ciphertext”, by means of an encryption algorithm.

Endpoint
A Universal Resource Indicator (URI) that exposes a service to users of the TOE.

Event Log
A record of security-relevant events recorded by the TOE. Authorized administrators can configure the categories of events the TOE records and the level of detail at which it records them.

FIPS PUB 140-2
Federal Information Processing Standards Publication 140-2, titled “Security Requirements for Cryptographic Modules.” This document specifies requirements for validation testing of cryptographic modules, such as key sizes they must support in order to receive a particular security level rating.

GUI
Graphical User Interface; a human interface that maps computer functions to graphical objects that the user can manipulate by means of a pointing device to perform tasks. Contrast with command-line interface, which requires the user to type text-based commands to perform tasks.

Handler
The component of the Web Services SFP that manages communication with consumers. When an incoming message meets all requirements imposed by an authenticator, the message is eligible for further processing by a handler that is a member of the authenticator's authorization group. The message is not available to handlers outside of the authorization group. A handler specifies the message protocol and network endpoint/port on which the Gateway accepts message traffic, as well as various criteria the incoming message must meet in order to be eligible for further processing by the Web Services SFP. A handler also passes a response from a protected service back to the consumer that made the original request, again subject to all requirements of the Web Services SFP. The ACE XML Manager GUI provides a graphical representation of each handler in the Web Services SFP. Authorized administrators interact with the Manager GUI to create, delete, or modify handlers or other policy objects that define the Web Services SFP.

Hash
A fixed string of numbers generated by applying a cryptographic algorithm to source text; the process of creating a hash is known as “hashing.”

HTTP Basic authentication
A method of identifying and authenticating the sender of a message by means of a username/password pair passed as plaintext in the header of an HTTP request message. Because of its susceptibility to eavesdropping, HTTP Basic authentication is more secure when used over an SSL connection.

HTTP header
A text record sent at the beginning of an HTTP or HTTPS message. Request message headers provide information about the client to the server receiving the request, such as the type of browser being used. In addition to information the header is required to provide, it may also include optional values such as the HTTP Basic username and password of the sender. Response message headers provide information from the server to the client that made the original request; for example, a response message may contain an error code that attempts to explain the reason a request did not succeed.

HTTP(S)
A typographical convention the TOE user interface and documentation uses to indicate use of either of the HTTP or HTTPS protocols. The HTTPS (HyperText Transfer Protocol Secure) protocol is the HTTP protocol

conducted in a session managed by a security protocol, such as SSL or TLS.

IBM MQ
The MQ Series message queue protocol defined by International Business Machines, Inc.

JMS
Java Messaging Service, a programming interface from Sun Microsystems that Java programs can use to communicate with message queue servers.

LDAP
Lightweight Directory Access Protocol, an Internet standard for data formats and protocols used to serve information about users, groups, addresses, and authentication.

Local authentication mode
The “Standard ACE XML Manager Passwords” authentication mode. Once within the boundaries of the TOE, the login data is authenticated by the TOE only, without use of any external services; hence, the name “local” authentication mechanism. Although the Manager supplies alternative mechanisms for authenticating administrative logins, only “local” (also called Standard) authentication mode is the evaluated configuration. Manager user accounts that rely on this authentication mechanism are sometimes referred to as “local” accounts.

MAC
See Message Authentication Code.

Manager Audit Log
A record of TOE user activity recorded by the TOE. The Manager Audit log is visible only to the special, factory-configured “administrator” account in the ACE XML Manager's Web-based graphical user interface. Because this log is intended to support security audits of the TOE, it is not configurable and always records the activities of every user of the TOE. This log records successful and failed login attempts; additions, deletions, and changes to Web Services SFPs, including consumer security attributes; deployment of Web Services SFPs to Gateways; startup and shutdown of Gateway I/O processes; and the startup and shutdown of each Gateway and Manager instance in the cluster.

Message Authentication Code (MAC)
A value computed from the contents of a message and a secret key. The SSL and TLS protocols' use of MACs ensures the authenticity of the sender and the integrity of a message, and prevents tampering with encrypted data.

Message Traffic Log
A record of messages processed by the ACE XML Gateway. Authorized administrators can configure the level of detail this log records, from statistics only through a record of the complete body of each message processed.

Message queue server
A network service that receives messages from consumers and delivers them simultaneously to one or more recipients. A message queue server provides a means of delivering information simultaneously to dissimilar platforms such as IBM, Windows, VMS, Java, and various UNIX platforms.

Password
A value used to authenticate a client to a server. Because this value is known only to the specific client and server conducting the transaction, the server can consider receipt of the correct password as proof of the identity of the client.

Policy
The data used to define and configure the rules that the ACE XML Gateway enforces to shield protected services against attack or misuse.

Port
A transport-specific value that associates a data packet sent over a network with a local process on the host that accepts the packet.

Reactor

An alternative message-processing engine built into the Gateway. Use of the “Reactor” engine is excluded from the evaluated configuration. Authorized administrators enable use of this engine on a per-HTTP-port basis (optionally, restricted to requests to a specific hostname or IP address) by checkmarking the “Always Use Flex Path” option on the HTTP port's **Edit Port** page in the Manager GUI; in the evaluated configuration, this box must not be checkmarked.

RFC

Request For Comments - A document that describes the specifications for a recommended technology. Although the word "request" is in the title, if the specification is ratified, it becomes a standards document. Not all RFCs become standards; some are designated indefinitely with Informational or Experimental status. The TOE was developed based the following RFCs in support of the security attributes that are used to enforce the information flow policies. RFC 986 (supports destination address attribute), RFC 2246 (supports Distinguished Name attribute), RFC 2617 (supports HTTP Authentication: Basic attribute), RFC 2616 (supports Request-URI attribute and GET attribute), RFC 2459 and 2246 (supports X.509 attribute), RFC 1867 (supports POST attribute).

RPC method name

In an RPC-style SOAP message, the URI of the SOAP service (the Remote Procedure Call) that the message invokes.

RSA

Rivest-Shamir-Adleman (RSA), a cryptographic algorithm by RSA Security, Inc., Bedford, MA. The RSA algorithm can be used to encrypt and decrypt data, as well as to generate digital signatures used to verify the authenticity of data. The "RSA" acronym itself comprises the initials of the last names of the inventors of this suite: Ron **R**ivest, Adi **S**hamir, and Leonard **A**dleman.
RSA is a very strong cryptography method but a computationally-intensive one; to optimize performance, it may be implemented in hardware, as it is by the nCipher hardware cryptomodule the IT environment of the TOE supplies. The FIPS certificate numbers for the nCipher 4000 PCI are 669 and 670.

RSA key generation algorithm

The method of generating a cryptographic key that the RSA public-key cryptosystem Secure Hash Algorithm defines in the RSA-SHA1 Signature Suite - Version 1.0. The ACE XML Manager offers the use of this algorithm as one means by which it can generate a cryptographic key.

SAML

Security Assertion and Markup Language, an XML-based language for encoding assertions that describe authentication and authorization. Authorized administrators can configure the Web services SFP to use SAML assertions to authenticate the sender of a SOAP message or to validate specific XML elements in its content.

SAML Assertion Validation

A means of validating message content by requiring it to contain specified SAML assertion elements. Authorized administrators can configure the Web services SFP to use this means of authenticating the sender of a SOAP message or to validate specific XML elements in its content. In addition to the RFCs and Schemas, the TOE also supports the following assertion; Assertions and Protocol for the OASIS Security Assertion Markup Language (SAML) V1.1 OASIS Standard, 2 September 2003.

Schema

An XML document that specifies the structure and usage of other XML documents. For the XML document type it defines, an XML schema specifies the tags the document may contain, their attributes, the order in which elements defined with those tags may appear, and the other elements each element may contain. An XML schema document can be used to determine whether an instance of the XML document type it describes is valid: whether the structure of the document instance conforms to the rules the schema specifies, and whether the content of the document contains values that are valid according to those rules. In addition to the RFCs, the TOE also supports the following schemas; XML Schema Part 0: Primer Second Edition W3C Recommendation 28 October 2004, XML Schema Part 1: Structures Second Edition, W3C Recommendation 28 October 2004, XML Schema Part 2: Datatypes Second Edition W3C Recommendation 28 October 2004 and XML-Signature Syntax and Processing W3C Recommendation 12 February 2002.

Service
A collection of related ports and the operations they support, exposed to TOE users as a Universal Resource Indicator (URI).

Service descriptor
A code object implementing that portion of the Web services SFP which manages communication with Web services the Gateway protects. A service descriptor accepts a request message from a handler by means of a route and passes the message to a Web service the Gateway protects; at the option of the authorized administrator configuring the service descriptor, it can validate or transform the message before sending it to the service. The service descriptor also accepts the response message from the Web service, optionally validating or transforming it before passing it back to the handler that sent the original request. The ACE XML Manager GUI provides a graphical representation of each service descriptor in the Web services SFP. Authorized administrators interact with the Manager GUI to create, delete, or modify service descriptors.

Session key
The cryptographic key that protects a secure connection, such as an SSL or TLS session.

SGML
Standard Generalized Markup Language, an ISO-standard language for describing data formats, based on IBM's Generalized Markup Language. XML is an SGML data format which is itself capable of describing still other data formats.

Signature
Digital signature, a unique value computed by using a cryptographic hashing function to apply a randomly-generated key to the contents of a message. A digital signature can be used to prove the identity of the sender of a message, and to determine whether the message was altered after it was signed. The TOE can verify digital signatures to establish the authenticity of digital certificates, the integrity of messages, and the identities of clients and servers.

SMTP
Simple Mail Transfer Protocol, a means of sending text messages and attachments over TCP/IP.

SOAP
Simple Object Access Protocol, an XML-based standard for making remote procedure calls by means of text messages, using HTTP(S) as the transport mechanism. In addition to the RFCs, schemas, and assertions, the TOE also supports the following SOAP versions, Simple Object Access Protocol (SOAP) 1.1 W3C Note 08 May 2000,SOAP Version 1.2 Part 2: Adjuncts W3C Recommendation 24 June 2003, and SOAP Version 1.2 Part 1: Messaging Framework (Second Edition) W3C Recommendation 27 April 2007.

SOAP Action
An element of a SOAP message that specifies the URI of the SOAP service that the message invokes.

SSL

Secure Sockets Layer, a secure protocol used to validate the identities of participants in a transaction and create an encrypted connection over which the transaction can take place.

TCP/IP

Transmission Control Protocol/Internet Protocol, a communications protocol for exchanging data between dissimilar systems as IP packets guaranteed to arrive intact and in correct order. TCP/IP is the de facto standard for global communications over the Internet.

TIB/RV

TIBCO Rendezvous, The Information Bus/Rendezvous, messaging software from TIBCO Software, Inc., that provides message bus connectivity to a variety of enterprise applications and platforms.

Timestamp

A value that records the time of day. Each log entry the TOE records includes a timestamp that indicates the time at which the TOE recorded the entry. Authorized administrators of the TOE can examine timestamps in log entries to determine the time at which a logged event occurred.

TLS

Transport Layer Security, an Internet Engineering Task Force security protocol based on SSL v.3.0 that protects data by securing communications channels. TLS encrypts all traffic on a specific channel, which is established using a cryptographically-secure protocol that prevents attackers from obtaining the data or the keys used to encrypt it.

URI

Uniform Resource Indicator, a standardized means of identifying resources on the Internet or on a private intranet.

Username

The value that identifies an account, such as one used to log into the ACE XML Manager. For the evaluated configuration, one must present a valid username and the correct password associated with that username in order to log into the ACE XML Manager.

X.509

An International Telecommunications Union standard that defines the format of a digital certificate.

X.509 certificate authentication

Presenting a digital certificate based on the X.509 standard as a means of proving one's identity. The TOE uses X.509 certificates for a variety of authentication functions. Authorized administrators can configure Web services SFPs to require a consumer to present an X.509 certificate to the Gateway as a prerequisite to sending a message. The TOE also requires bilateral exchange of X.509 certificates to establish communications between the Manager and the Gateway. Additionally, the TOE uses an X.509 certificate to encrypt the connection over which such communications take place. Furthermore, the Manager presents an X.509 certificate to authenticate itself to the Web browser from which an authorized administrator logs into the Manager.

XML

Extensible Markup Language, a flexible formal text format derived from SGML that is commonly used to define more specialized markup languages for representing structured data.

XML Schema Validation

The use of an XML schema document to test the validity of the structure or content of an XML document of the type the schema describes. *See also* Schema.

XML Signature Verification A method of establishing the authenticity of a document or its sender by using a shared secret (key) to recompute a cryptographic digest computed from the contents of the document or the certificate the sender presents. If the two signatures match, the document or sender is authentic.

XSD The type identifier that identifies an element of an XML Schema Document. *See also* Schema.

2 TOE Description

The Target of Evaluation (TOE) is the Cisco Systems, Inc. ACE XML Gateway and Manager Version 5.0.3.

The Cisco Systems, Inc. ACE XML Gateway and Manager Version 5.0.3 is a self-contained IT appliance that can be configured to run as a Cisco ACE XML Gateway, as a Cisco ACE XML Manager, or as both Gateway and Manager simultaneously. The evaluated configuration excludes the configuration that runs both the Manager and the Gateway simultaneously on a single ACE XML appliance.

- The **ACE XML Gateway** stands between an untrusted network (the Internet) and a trusted network (such as a restricted-access corporate intranet). All traffic between the two networks must pass through the Gateway. The Gateway allows only authorized traffic to pass from the untrusted network to the trusted network. Authorized administrators specify the criteria that traffic must meet in order to pass through the Gateway. The Gateway blocks traffic that does not meet these criteria. The Gateway generates an audit trail that documents the performance of the Gateway, the disposition of every message it processes, and other security-relevant events.
- The **ACE XML Manager** provides a graphical user interface (GUI) that authorized administrators use to specify the message-processing behavior of the Gateway, monitor the performance of the Gateway, and manage the Gateway remotely. The Manager GUI provides a means of viewing the audit trail generated by all Gateways in the scope of the Manager's control and the activities of the users of the Manager.

The ACE XML Gateway (Gateway) is an application running in the context of a custom version of Linux installed on a 1U chassis, which is a Hewlett-Packard DL360 G5 server hardware appliance with nCipher 4000 PCI cryptographic module. The ACE XML Manager (Manager) is a Java application running in the context of a custom deployment of the Tomcat application server, which in turn runs in the context of a custom version of Linux installed on a 1U chassis, which is a Hewlett-Packard DL360 G5 hardware server appliance with nCipher 4000 PCI cryptographic module.

The specific Linux components installed on each of the Manager and Gateway appliances are specified in the *Cisco Systems ACE XML Gateway and Manager Version 5.0.3 Configuration Management Procedures* document. The 5.0.3-2007-06-12T23 release identifier string in the TOE user interface identifies this software and firmware configuration required to create the evaluated configuration. The TOE IGS procedures describe how to verify the presence of this string and require the customer to do so.

The specific hardware components installed on each of the Manager and Gateway appliances are specified in the *Cisco Systems ACE XML Gateway and Manager Version 5.0.3 Delivery Procedures* document. The TOE IGS procedures describe how to verify the hardware configuration of each appliance and require the customer to do so.

The evaluated configuration requires one ACE XML appliance that runs as a dedicated ACE XML Manager only (the "Manager appliance"), and at least one additional ACE XML appliance that runs as a dedicated Gateway only (the "Gateway appliance"). One Manager can control multiple Gateway instances, thus allowing for scalability of the TOE. All communications between Manager and Gateway instances take place exclusively on the trusted network over a secure, encrypted connection.

All users of the TOE are either administrative or non-administrative. **Authorized administrators** are users that log into the Manager to perform administrative actions. All other TOE users are non-administrative. **Non-administrative users** are the holders of view-only accounts (External Developer, Policy View) on the Manager, or the consumers of proxy services the TOE provides (humans or processes that exchange messages and data with Web services the Gateway protects).

The Manager provides a graphical user interface (GUI) that Manager accountholders use to interact with the TOE. The Manager displays its GUI in a Web browser (Policy Administration Web Client) on a computer that the IT environment provides. In order to manage and monitor the Gateway, an authorized administrator must log on to an administrative user account on that Manager instance. Holders of non-administrative user accounts may also interact with the Manager GUI on a limited, view-only basis. Consumers of TOE services do not have Manager login accounts, never interact directly with the TOE, and may not even know that it exists.

To protect TSF data from disclosure while in transit between physically-separate parts of the TOE, SSL is used to provide a private, encrypted connection between the user's Web browser and the Manager, and between the Manager and the Gateway. The Manager communicates with its assigned Gateways in a proprietary, binary format on this trusted channel. For more information, see Section 6.1.3, "Identification and Authentication." And Section 6.1.5 "Protection of the TSF".

Users must identify and authenticate successfully to a local user account on the Manager in order to access any security-relevant services or functionality the TOE provides. The TOE makes no services or information available until the user identifies and authenticates successfully. Only authorized administrators and users that have been assigned appropriate security roles can use the Manager GUI to access TSFs. Authorized administrators must use the Manager GUI for creating and managing WEB SERVICES SFP security policies, for loading and running a security policy on Gateway instances, for managing users, and for monitoring the Gateway.

Using the graphical user interface to the ACE XML Manager, authorized administrators define a security function policy, the WEB SERVICES SFP. At the request of an authorized administrator, the ACE XML Manager loads the security function policy onto one or more ACE XML Gateway instances the Manager controls. Each ACE XML Gateway instance enforces the security function policy immediately.

The Manager polls its Gateway instances regularly for records of security-relevant events that the Manager appliance records on its own local storage as time-stamped entries in three logs that comprise the audit trail: the Event Log, the Message Traffic Log, and the Manager Audit Log. Only authenticated administrators and users that have been assigned appropriate security roles can view or query these logs in the Manager user interface.

2.1 TOE Overview

The TOE is an application-level proxy that processes XML and SOAP messages sent across TCP/IP networks using HTTP(S) protocols. XML is a flexible formal text format derived from SGML and commonly used to define more specialized markup languages for representing computer data. SGML is an ISO-standard language for describing data formats, based on IBM's Generalized Markup Language. SOAP is an XML-based protocol for making remote procedure calls by means of text messages, using HTTP(S) as the transport mechanism. The TOE is depicted in the figure below in the context of its location in the IT environment.

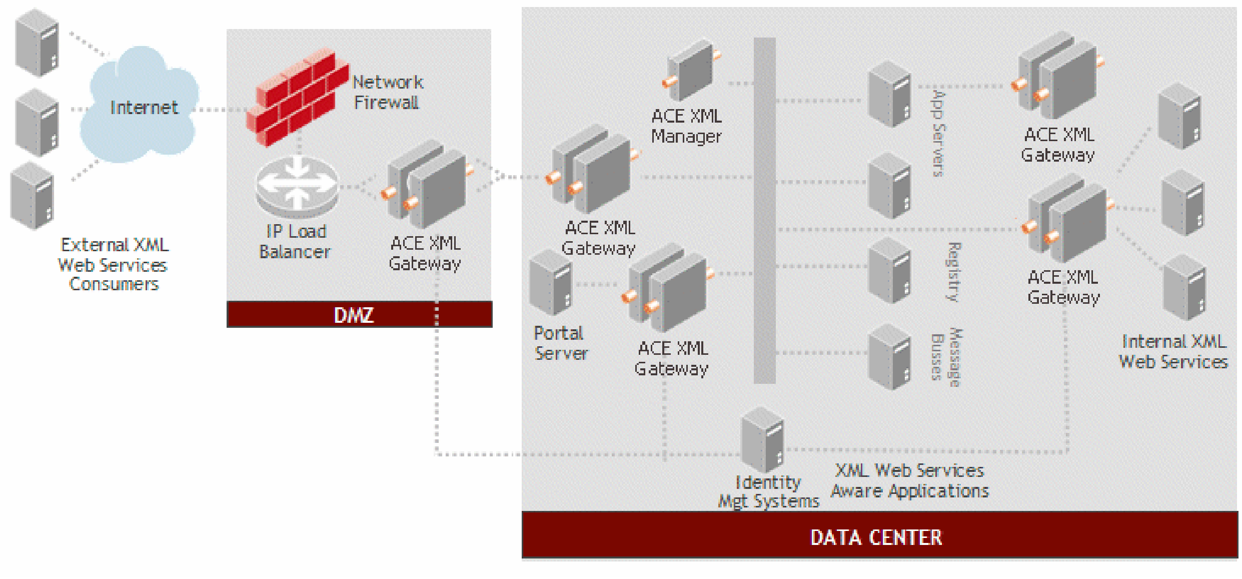

Figure 1: ACE XML Gateway and ACE XML Manager

The TOE cannot be bypassed; in order to reach the trusted network, traffic from the untrusted network must pass through the Gateway, subject to the rules the Web Services SFP defines.

The TOE provides a secure environment for the exchange and use of XML, ensuring no malicious content, no denial-of-service attacks, confidentiality and integrity of valuable and private messages, and appropriate access control for those services. XML exposes business logic as independent services (Web services and Service Oriented Architecture, specifically). In addition, the TOE may optionally be configured to provide persistent logging of messages, to interact with external authorization services, and to transform messages during processing.

2.1.1 Features That Are Outside the Scope of the Evaluated Configuration

The following list of features is not included in the evaluated configuration.

- Cryptography: Cryptographic functionalities are provided by the environment in the evaluated configuration.
- LDAP Support for Message Authentication and Authorization: In the Common Criteria evaluated configuration LDAP Support is not allowed for authentication and authorization of messages.
- Java SDK: In the Common Criteria evaluated configuration Java SDK customization or authorization logic is not allowed.
- Message Transformation: In the Common Criteria evaluated configuration transformations specified in the XSL language to messages are not allowed.
- Message Caching: In the Common Criteria evaluated configuration end-user specified message caching is not allowed.
- SNMP Monitoring: In the Common Criteria evaluated configuration SNMP monitoring is excluded.
- System Snapshot diagnostic tool: The use of the system snapshot functionality is not allowed in the Common Criteria evaluated configuration.
- Access Control: Sub-policies: The Common Criteria evaluated configuration does not allow the creation of and excludes the use of sub-policies other than the factory-configured "Shared" sub-policy.
- Access Control: Approval-Based Deployment: The Common Criteria evaluated configuration does not allow the approval-based deployment feature to be enabled.
- Access Control: Alternate Authentication Mechanisms: LDAP: LDAP authentication of Manager user accounts is not allowed in the Common Criteria evaluated configuration.
- Message Routing: Fast path Engine: The Common Criteria evaluated configuration excludes use of the “Reactor” (also known as the Fast Path) message-processing engine.
- Protocols: The Common Criteria evaluated configuration excludes the use of the SMTP, JMS, MQ or TIBCO message protocols for use with handlers and service descriptors.

See the Administrative guidance (*Using the Cisco Systems ACE XML Gateway and Manager Version 5.0.3*) for additional details.

2.2 TOE Architecture

The components that make up the TOE are:

- Gateway – The Gateway executable. Subject to the rules of the Web Services SFP, the Gateway proxies XML and SOAP messages sent across TCP/IP networks using HTTP(S) protocols. The Gateway application runs in the context of a custom version of Red Hat Enterprise Linux version 3, update 8 installed on a 1U chassis, which is a Hewlett-Packard DL360 G5 server hardware appliance with nCipher 4000 PCI cryptographic module.
- Manager – The Manager application. The Manager application provides a GUI that authorized administrators use to administer the Gateway application; in particular, authorized administrators use the

Manager GUI to define the Web Services SFP that the Gateway enforces. The Manager application runs in the context of an Apache Tomcat application server running in the context of a custom version of Linux installed on a 1U chassis, which is a Hewlett-Packard DL360 G5 server hardware appliance with nCipher 4000 PCI cryptographic module.

- Tomcat – Each ACE XML appliance embeds an Apache Tomcat v. 5.0.16 application server that the Manager uses to publish its Web-based GUI.
- Shell - A terminal-based program that runs automatically when an authorized administrator logs in to the console of a ACE XML Manager or Gateway machine. The Shell provides tools for low-level administration of ACE XML systems, such as changing network configuration.
- Operating system files – A number of operating system files are used by both the Gateway and the Manager for configuration and logging.
- Operating System – Each ACE XML appliance embeds a custom, package-reduced installation of the Linux operating system. This operating system runs on the server hardware chassis, hosting the TOE software and the Web server that publishes the Manager GUI.
- Server Hardware chassis – The ACE XML appliance is built on a Hewlett-Packard DL360 G5 server hardware chassis. This chassis hosts the Operating System, Application Server, TOE software/firmware and nCipher 4000 PCI cryptomodule. Note that although the cryptomodule resides physically on the server chassis, the ST considers this module to be provided by the IT environment because it is used ”off-the-shelf” with no modifications; see Section 6.1.5, “Protection of the TSF” for more information. For local storage, the server chassis provides two 72 GB hard drives configured as a RAID 1 array by the manufacturer of the chassis. The server chassis also has four physical Ethernet ports, and connections for a serial keyboard and VGA monitor.

2.2.1 Physical Boundaries

The TOE is installed on the following platforms:

- Linux operating system
- Tomcat application server on the Manager appliance
- Hewlett-Packard DL360 G5 server hardware appliance with nCipher 4000 PCI cryptomodule

2.2.2 Logical Boundaries

This section identifies the security functions that the TSF provides**.**

- Security Audit (FAU)
- User Data Protection (FDP)
- Identification and Authentication (FIA)
- Security Management (FMT)
- TSF Protection (FPT)

2.2.2.1 Security Audit (FAU)

The TOE generates audit events for the minimum level of audit. The TOE provides Manager GUI interfaces that can be used to read the audit trail. The TOE restricts access to the audit trail, requiring authentication using its local account authentication mechanism.

2.2.2.2 User Data Protection (FDP)

The TOE enforces the WEB SERVICES SFP on SOAP or HTTP(S) destination service traffic sent through the TOE from one consumer (subject) to another. The TOE enforces the WEB SERVICES SFP, using “authenticators” to verify the user and group identity of a consumer of a service, using “handlers” to validate incoming messages, using “routes” to pass accepted message to “service descriptors,” and using “service descriptors” to manage traffic with SOAP or HTTP(S) destination services according to the WEB SERVICES SFP configuration for a given Web service. The TOE supports multiple message-filtering mechanisms for use by the WEB SERVICES SFP depending on configuration for a given Web service. The TOE includes pluggable authentication modules that can call external authentication servers to verify the user and group identity of a consumer of a service for message-filtering purposes.

2.2.2.3 Identification and Authentication (FIA)

The TOE disables user or administrator accounts after three failed login attempts to the Manager. The TOE maintains user identities, authentication data for supported authentication mechanisms, and role information. The TOE offers no TSF-mediated functions until the user is authenticated. The TOE requires username/password for all user accesses to the Manager. The TOE offers no TSF-mediated functions until the user is identified.

2.2.2.4 Security Management (FMT)

The TOE restricts the ability to specify the Web Services SFP to authorized administrators. The TOE provides restrictive default values for security attributes used to enforce the WEB SERVICE SFP. The TOE also allows authorized administrators to specify alternative initial values. The TOE restricts the ability to initialize and set user authentication data to authorized administrators. The TOE restricts the ability to modify and reset an account’s own password to authorized administrators and users. The TOE restricts the ability to view or query audit records to authorized administrators or users that have been assigned appropriate security roles. The TOE provides authorized administrators with the ability to manage Web services, to manage users, and to manage the audit trail using the Manager. The TOE supports two types of users, authorized administrators and users. The single factory-configured `administrator` account always has all security roles (in particular, the ConsoleAdmin role), cannot be modified or deleted, and is considered an “authorized administrator”. The second category of administrative user is a user that has been assigned zero or more system-defined roles. The system-defined roles are “Operations”, “Access Control”, MTL (message traffic log), or “Routing”. is considered an “authorized administrator” and any other user accounts are considered simply “users.” The non-administrative or user category comprises view-only accounts (External Developer and Policy View) on the Manager.

2.2.2.5 Protection of the TSF (FPT)

The TOE can generate reliable time stamps for its own use. The TOE can send handler test messages in order to demonstrate the correct operation of a configured handler, route, service descriptor, Web service, and the underlying network. The TOE can also test its network configuration in order to demonstrate its correct configuration. The TOE uses SSL when managing the Gateway using the Manager to protect TSF data from disclosure. The TOE protects against denial-of-service attacks by blocking traffic after administratively-configurable thresholds are met. The TOE protects against content-based attacks by rejecting messages that contain content marked as blocked. The WEB SERVICES SFP cannot be bypassed by consumers. Similarly, both Gateway and Manager interfaces are restricted to authorized administrators and user account-holders.

Upon startup, the TOE enters a restrictive default state in which no users are logged in, and then resumes normal operation. Because the TOE cannot be bypassed, this default state is secure: the Gateway enforces the current Web Services SFP independently of the Manager, the Gateway accepts changes to the current Web Services SFP only from its Manager, and the user interface to the Manager provides no access to TSFs until the user identifies and authenticates successfully.

2.3 TOE Documentation

Cisco Systems offers a series of documents that describe the installation process for the Gateway and Manager applications as well as guidance for subsequent use and administration of the applicable security features. Refer to Section 6 for information about these and other documents associated with the TOE.

3 Security Environment

This section summarizes the threats and organizational security policies addressed by the TOE and assumptions about the intended environment of the TOE. Note that while the identified threats are mitigated by the security functions implemented in the TOE, the overall assurance level (EAL 3 augmented with ALC_FLR.2) also serves as an indicator of whether the TOE would be suitable for a given environment.

3.1 Threats

T.SELPRO The TOE must protect itself against attempts by unauthorized users to bypass, deactivate, or tamper with TOE security functions.

3.2 Organizational Security Policies

P.ACCOUN The TOE must provide user accountability for information flows through the TOE and for all use of security functions related to audit.

P.AUDREC The TOE must provide a readable audit trail of security-related events.

P.ENCRYP The TOE must protect the confidentiality of its dialogue with an authorized administrator through encryption, if the TOE allows administration to occur remotely from a connected network.

P.IDAUTH The TOE must uniquely identify and authenticate the claimed identity of all users, before granting a user access to TOE functions or, for certain specified services, to a connected network.

P.SECFUN The TOE must provide functionality that enables an authorized administrator or user with appropriate security roles to use the TOE security functions, and must ensure that only authorized administrators *or users with appropriate security roles* are able to access such functionality.

P.LIMEXT The TOE must provide the means for an authorized administrator to control and limit access to TOE security functions by an authorized external IT entity.

3.3 Secure Usage Assumptions

3.3.1 Intended Usage Assumptions

A.ACCESS The TOE has access to all the IT System data it needs to perform its functions.

A.ASCOPE The TOE is appropriately scalable to the IT System the TOE monitors.

A.DYNMIC The TOE will be managed in a manner that allows it to appropriately address changes in the IT System the TOE monitors.

A.SINGEN Information can not flow among the internal and external networks unless it passes through the TOE.

3.3.2 Physical Assumptions

A.LOCATE The processing resources of the TOE will be located within controlled access facilities, which will prevent unauthorized physical access.

A.PROTCT The TOE hardware and software critical to security policy enforcement will be protected from unauthorized physical modification.

3.3.3 Personnel Assumptions

A.MANAGE There will be one or more competent individuals assigned to manage the TOE and the security of the information it contains.

A.NOEVIL The authorized administrators and users are not careless, willfully negligent, or hostile, and will follow and abide by the instructions provided by the TOE documentation.

4 Security Objectives

This section summarizes the security objectives for the TOE and its environment.

4.1 Security Objectives for the TOE

O.ACCOUN The TOE must provide user accountability for all attempts at authentication, for all attempted information flows through the TOE, for all attempted use of TSF-mediated security-management or audit functions, and for all modifications to the values of TSF data.

O.AUDREC The TOE must provide a readable audit trail of security-related events.

O.IDAUTH The TOE must uniquely identify and authenticate the claimed identity of all users, before granting a user access to TOE functions or, for certain specified services, to a connected network.

O.SECFUN The TOE must provide functionality that enables an authorized administrator *or user with appropriate security roles* to use the TOE security functions, and must ensure that only authorized administrators *or users with appropriate security roles* are able to access such functionality.

O.SELPRO The TOE must protect itself against attempts by unauthorized users to bypass, deactivate, or tamper with TOE security functions.

O.LIMEXT The TOE must provide the means for an authorized administrator to control and limit access to TOE security functions by an authorized external IT entity.

O.TIME The TOE must provide reliable time stamps.

4.2 Security Objectives for the IT Environment

OE.ENCRYP The IT environment will protect the confidentiality of its dialogue with an authorized administrator through encryption when performing remote administration from a connected network.

4.3 Security Objectives for the Environment

OE.INSTAL Those responsible for the TOE must ensure that the TOE is delivered, installed, managed, and operated in a manner which is consistent with IT security.

OE.PHYCAL Those responsible for the TOE must ensure that those parts of the TOE critical to security policy are protected from any physical attack.

OE.CREDEN Those responsible for the TOE must ensure that all key file passphrases and key files are protected by the users in a manner which is consistent with IT security.

OE.PERSON Personnel working as authorized administrators shall be carefully selected and trained for proper operation of the System.

OE.INTROP The TOE has the necessary interactions with and access to the IT System it monitors.

OE.SINGEN The information cannot flow between internal and external networks without flowing through the TOE, and this configuration is protected by physical protection of the TOE interfaces.

5 IT Security Requirements

This section defines the security functional requirements for the TOE as well as the security assurance requirements against which the TOE has been evaluated. All of the requirements have been copied from version 2.2 of the applicable Common Criteria documents.

5.1 TOE Security Functional Requirements

The following table describes the SFRs that are satisfied by the Cisco Systems, Inc. ACE XML Gateway and Manager Version 5.0.3.

Requirement Class	Requirement Component
FAU: Security Audit	FAU_GEN.1: Audit data generation
	FAU_SAR.1.1a,b,c: Audit review : admin vs. non-admin
	FAU_SAR.1.2: Audit review : presentation
	FAU_STG.1: Protected audit trail storage
FDP: User data protection	FDP_IFC.1: Subset information flow control
	FDP_IFF.1: Simple security attributes
FIA: Identification and authentication	FIA_AFL.1: Authentication failure handling
	FIA_ATD.1: User attribute definition
	FIA_UAU.2: User authentication before any action
	FIA_UID.2: User identification before any action
FMT: Security management	FMT_MOF.1: Management of security functions behavior
	FMT_MSA.1.1a,b,c: Management of security attributes
	FMT_MSA.3: Static attribute initialization
	FMT_MTD.1.1a,b,c,d, e: Management of TSF data
	FMT_REV.1: Revocation
	FMT_SMF.1: Specification of Management Functions
	FMT_SMR.1: Security roles
FPT: Protection of the TSF	FPT_ITT.1: Basic internal TSF data transfer protection
	FPT_RVM.1: Non-bypassability of the TSP
	FPT_STM.1: Reliable time stamps

Table 1: TOE Security Functional Components

5.1.1 Security Audit (FAU)

5.1.1.1 Audit data generation (FAU_GEN.1)

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 **[*minimum*]** level of audit; and

c.) **[The following auditable events:**

i. Message traffic or information about message traffic passing between consumers and services[1]

ii. All successful or unsuccessful attempts at modification of TSF data]

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

[1] See Section 6.1.1 for the details about the information that gets recorded for message traffic.

b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, [**no other audit relevant information**].

5.1.1.2 Audit review (FAU_SAR.1)

FAU_SAR.1.1a The TSF shall provide **[authorized administrators]** with the capability to read **[all audit information]** from the audit records.

FAU_SAR.1.1b The TSF shall provide **[users in the Policy View role]** with the capability to read **[Event Log information]** from the audit records.

FAU_SAR.1.1c The TSF shall provide **[users in the Message Traffic Log role]** with the capability to read **[Message Traffic Log information]** from the audit records.

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

5.1.1.3 Protected audit trail storage (FAU_STG.1)

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

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

5.1.2 User data protection (FDP)

5.1.2.1 Subset information flow control (FDP_IFC.1)

FDP_IFC.1.1 The TSF shall enforce the **[WEB SERVICES SFP]** on:

a.) **[subjects: human user or external IT entity (consumer or Web service) that sends and receives HTTP(S) destination service information through the TOE to another subject, only after the subject initiating the information flow (consumer) has authenticated at the TOE per FDP_IFF.1;**

b.) **information: SOAP, arbitrary XML, or arbitrary raw data sent through the TOE from one subject to another; and**

c.) **operation: initiate service and pass information]**

5.1.2.2 Simple security attributes (FDP_IFF.1)

FDP_IFF.1.1 The TSF shall enforce the **[WEB SERVICES SFP]** based on at least the following types of subject and information security attributes:

a.) **[subject security attributes:**

i. **presumed address (consumer's)**

b.) **information security attributes:**

i. **Destination address**
ii. **User name and password (consumer)**
iii. **SSL Client Certificate Distinguished Name Parameter (consumer credential attributes)**
iv. **XML Schema Validation**
v. **XML Signature Verification**
vi. **SAML Assertion Validation**
vii. **Port on which a handler accepts messages**
viii. **Local endpoint descriptor a handler exposes to consumers**
ix. **Version of SOAP to use for transactions**
x. **Requested URI**
xi. **Requested RPC method name**
xii. **Requested SOAP Action HTTP Header Value**
xiii. **X.509 certificate(consumer credential attributes)]**

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

a.) **[Subjects on an internal network can cause information to flow through the TOE to another connected network if:**

   i. **all the information security attribute values the information provides are permitted unambiguously by rules an authorized administrator composes as combinations of information flow security attribute values the rules require.**

b.) **Subjects on the external network can cause information to flow through the TOE to another connected network if:**

   i. **all the information security attribute values the information provides are permitted unambiguously by rules an authorized administrator composes as combinations of information flow security attribute values the rules require.]**

FDP_IFF.1.3 The TSF shall enforce the **[none]**.

FDP_IFF.1.4 The TSF shall provide the following **[none]**.

FDP_IFF.1.5 The TSF shall explicitly authorize an information flow based on the following rules: **[none]**.

FDP_IFF.1.6 The TSF shall explicitly deny an information flow based on the following rules:

a.) **[The TOE shall reject requests where all parts of messages intended for the requested Web service are not included]**.

b.) **[The TOE shall reject request messages that contain the following content marked as blocked: any of the SQL commands ALTER DATABASE, ALTER TABLE, ALTER VIEW, CREATE DATABASE, CREATE PROCEDURE, CREATE SCHEMA, CREATE TABLE, CREATE VIEW, DELETE FROM, DROP DATABASE, DROP TABLE, DROP VIEW]**.

c.) **[The TOE shall reject request messages that contain the following content marked as blocked: any SQL Server 2000 Master database default table names]**.

5.1.3 Identification and authentication (FIA)

5.1.3.1 Authentication failure handling (FIA_AFL.1)

FIA_AFL.1.1 The TSF shall detect when [**an administrator configurable setting of**] unsuccessful authentication attempts occur related to **[account access].**

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been met or surpassed, the TSF shall **[disable the account]**.

5.1.3.2 User attribute definition (FIA_ATD.1)

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

a) [user identity

b) authentication data (password)

c) role(s)].

5.1.3.3 User authentication before any action (FIA_UAU.2)

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.

5.1.3.4 User identification before any action (FIA_UID.2)

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.

5.1.4 Security management (FMT)

5.1.4.1 Management of security functions behavior (FMT_MOF.1)

FMT_MOF.1.1 The TSF shall restrict the ability to **[*disable and enable*]** the functions **[setting Web services policies]** to **[authorized administrator]**.

5.1.4.2 Management of security attributes (FMT_MSA.1)

FMT_MSA.1.1a The TSF shall enforce the **[*access control SFP*]** to restrict the ability to **[*modify, delete*]** the security attributes **[subject security attributes, information security attributes, user security attributes]** to **[administrator (all attributes) or user that has been assigned to the Access Control role (information security attributes only), Routing role (information security attributes only), or Operations role (information security attributes only)** [2]**]**.

FMT_MSA.1.1b The TSF shall enforce the **[*access control SFP*]** to restrict the ability to **[[view]]** the security attributes **[subject security attributes]** to **[administrator or user that has been assigned to the Policy View role]**.

FMT_MSA.1.1c The TSF shall enforce the **[*access control SFP*]** to restrict the ability to **[[view]]** the security attributes **[information security attributes]** to **[authorized administrators or user that has been assigned to the External Developer role or Policy View role]**.

5.1.4.3 Static attribute initialization (FMT_MSA.3)

FMT_MSA.3.1 The TSF shall enforce the **[WEB SERVICES SFP]** to provide **[*restrictive*]** default values for security attributes that are used to enforce the SFP. *(per International Interpretations #201 and #202)*

FMT_MSA.3.2 The TSF shall allow the **[authorized administrator**[3]**]** to specify alternative initial values to override the default values when an object or information is created.

5.1.4.4 Management of TSF data (FMT_MTD.1)

FMT_MTD.1.1a The TSF shall restrict the ability to **[[initialize, set]]** the **[user authentication data]** to **[authorized administrator**[4]**]**.

FMT_MTD.1.1b The TSF shall restrict the ability to **[*modify*, [reset]]** the **[user authentication data (password) of the user's own account]** to **[authorized administrators and users]**.

FMT_MTD.1.1c The TSF shall restrict the ability to **[*query*, [view]]** the **[audit records]** to **[authorized administrator**[5]**]**.

FMT_MTD.1.1d The TSF shall restrict the ability to **[*query*]** the **[audit records (Event Log)]** to **[authorized administrator or user that has been assigned to the Policy View role]**.

FMT_MTD.1.1e The TSF shall restrict the ability to **[*query*]** the **[audit records (Message Traffic Log)]** to **[administrator or user that has been assigned to the Message Traffic Log role]**.

5.1.4.5 Revocation (FMT_REV.1)

FMT_REV.1.1 The TSF shall restrict the ability to revoke security attributes associated with the **[*subjects*]** within the TSC to **[authorized administrators]**. *(per International Interpretation #201)*

FMT_REV.1.2 The TSF shall enforce the rules **[the enforcement of subject attribute changes shall take effect immediately on completion of the revocation operation]**.

5.1.4.6 Specification of Management Functions *(per International Interpretation #65)* (FMT_SMF.1)

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

a.) **Manage Web services**

b.) **Manage users**

c.) **Manage the audit trail]**. *(per International Interpretation #65)*

[2] See Section 6.1.4 for information on the roles and their associated privileges.

[3] A user assigned to the 'administrator' role. See Section 6.1.4 for information on the roles and their associated privileges.

[4] A user assigned to the 'administrator' role. See Section 6.1.4 for information on the roles and their associated privileges.

[5] A user assigned to the 'administrator' role. See Section 6.1.4 for information on the roles and their associated privileges.

5.1.4.7 Security roles (FMT_SMR.1)

FMT_SMR.1.1 The TSF shall maintain the roles **[authorized administrator, and users[6]]**.

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

5.1.5 Protection of the TSF (FPT)

5.1.5.1 Basic internal TSF data transfer protection (FPT_ITT.1)

FPT_ITT.1.1 The TSF shall protect TSF data from **[*disclosure*]** when it is transmitted between physically-separated parts of the TOE.

5.1.5.2 Non-bypassability of the TSP (FPT_RVM.1)

FPT_RVM.1.1 The TSF shall ensure that TSP enforcement functions are invoked and succeed before each function within the TSC is allowed to proceed.

5.1.5.3 Reliable time stamps (FPT_STM.1)

FPT_STM.1.1 The TSF shall be able to provide reliable time stamps for its own use.

[6] See Section 6.1.4 for information on the roles and their associated privileges.

5.2 IT Environment Security Functional Requirements

The following table describes the SFRs that are candidates to be satisfied by the IT environment of the TOE.

Requirement Class	Requirement Component
FCS: Cryptographic support	FCS_CKM.1: Cryptographic key generation
	FCS_COP.1: Cryptographic operation

Table 2: IT Environment Security Functional Components

5.2.1 Cryptographic support (FCS)

5.2.1.1 Cryptographic key generation (FCS_CKM.1)

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with the specified cryptographic key generation algorithm **[listed below]** and specified cryptographic key sizes **[specified for each algorithm]** that meet the following: **[standards noted for each algorithm:**

a.) **RSA key generation algorithm [*1024 bits*, *2048 bits*] as specified in PKCS#1**
b.) **DSA [*1024 bits*, *2048 bits*] bits as specified in FIPS-186**
c.) **3DES 192 bits (ANSI X9.52)**
d.) **AES [*128 bits*, *192 bits*, *256 bits*] (FIPS 197)**
e.) **Session key generation 192 bits as specified in SSL v3**
f.) **Session key generation 192 bits as specified in TLS v1].**

5.2.1.2 Cryptographic operation (FCS_COP.1)

FCS_COP.1.1 The TSF shall perform **[**

a.) **Signature generation**
b.) **Signature verification**
c.) **Hash generation**
d.) **SSL**
e.) **Encryption**
f.) **Decryption]**

in accordance with a specified cryptographic algorithm **[listed below]** and cryptographic key sizes **[specified for each algorithm]** that meet the following: **[standards noted for each algorithm:**

a.) **Signature generation/verification RSA [*1024 bits*, *2048 bits*] (PKCS #1, FIPS PUB 186-2, and ANSI X9.31) with SHA-1 160 bits (FIPS PUB 180-1, ANSI X9.30 Part 2), DSA [*1024 bits*, *2048 bits*] (FIPS PUB 186-2, ANSI X9.30) with SHA-1 160 bits**
b.) **Hash generation SHA-1 160 bits (FIPS PUB 180-1, ANSI X9.30 Part 2)**
c.) **Secure Sockets Layer (SSL) 3DES 192 bits (IETF SSL v3.0)**
d.) **Encryption/Decryption 3DES 192 bits (ANSI X9.52), AES [*128 bits*, *192 bits*, *256 bits*] (FIPS 197)].**

5.3 TOE Security Assurance Requirements

The security assurance requirements for the TOE are the EAL 3 augmented with ALC_FLR.2 components as specified in Part 3 of the Common Criteria. No operations are applied to the assurance components.

Requirement Class	Requirement Component
ACM: Configuration management	ACM_CAP.3: Authorization controls ACM_SCP.1 TOE CM Coverage
ADO: Delivery and operation	ADO_DEL.1: Delivery procedures ADO_IGS.1: Installation, generation, and start-up procedures
ADV: Development	ADV_FSP.1: Informal functional specification ADV_HLD.2: Security enforcing high-level design ADV_RCR.1: Informal correspondence demonstration
AGD: Guidance documents	AGD_ADM.1: Administrator guidance AGD_USR.1: User guidance
ALC: Life cycle support	ALC_FLR.2: Flaw reporting procedures ALC_DVS.1: Identification of Security Measures
ATE: Tests	ATE_COV.2: Analysis of coverage ATE_DPT.1: Testing: high-level design ATE_FUN.1: Functional Testing ATE_IND.2: Independent testing – sample
AVA: Vulnerability assessment	AVA_SOF.1: Strength of TOE security function evaluation AVA_VLA.1: Developer vulnerability analysis AVA_MSU.1 Examination of guidance

Table 3: EAL 3 augmented with ALC_FLR.2 Assurance Components

5.3.1 Configuration management (ACM)

5.3.1.1 Authorization Controls (ACM_CAP.3)

ACM_CAP.3.1D The developer shall provide a reference for the TOE.
ACM_CAP.3.2D The developer shall use a CM system.
ACM_CAP.3.3D The developer shall provide CM documentation.
ACM_CAP.3.1C The reference for the TOE shall be unique to each version of the TOE.
ACM_CAP.3.2C The TOE shall be labelled with its reference.
ACM_CAP.3.3C The CM documentation shall include a configuration list and a CM plan.
ACM_CAP.3.4C The configuration list shall uniquely identify all configuration items that comprise the TOE.
ACM_CAP.3.5C The configuration list shall describe the configuration items that comprise the TOE.
ACM_CAP.3.6C The CM documentation shall describe the method used to uniquely identify the configuration items that comprise the TOE.
ACM_CAP.3.7C The CM system shall uniquely identify all configuration items that comprise the TOE.
ACM_CAP.3.8C The CM plan shall describe how the CM system is used.
ACM_CAP.3.9C The evidence shall demonstrate that the CM system is operating in accordance with the CM plan.
ACM_CAP.3.10C The CM documentation shall provide evidence that all configuration items have been and are being effectively maintained under the CM system.
ACM_CAP.3.11C The CM system shall provide measures such that only authorised changes are made to the configuration items.

ACM_CAP.3.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.1.2 TOE CM Coverage (ACM_SCP.1)

ACM_SCP.1.1D The developer shall provide a list of configuration items for the TOE.

ACM_SCP.1.1C The list of configuration items shall include the following: implementation representation and the evaluation evidence required by the assurance components in the ST.

ACM_SCP.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.2 Delivery and operation (ADO)

5.3.2.1 Delivery procedures (ADO_DEL.1)

ADO_DEL.1.1d The developer shall document procedures for delivery of the TOE or parts of it to the user.

ADO_DEL.1.2d The developer shall use the delivery procedures.

ADO_DEL.1.1c The delivery documentation shall describe all procedures that are necessary to maintain security when distributing versions of the TOE to a user's site.

ADO_DEL.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.2.2 Installation, generation, and start-up procedures (ADO_IGS.1)

ADO_IGS.1.1d The developer shall document procedures necessary for the secure installation, generation, and start-up of the TOE.

ADO_IGS.1.1c The installation, generation and start-up documentation shall describe all the steps necessary for secure installation, generation and start-up of the TOE.

ADO_IGS.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADO_IGS.1.2e The evaluator shall determine that the installation, generation, and start-up procedures result in a secure configuration.

5.3.3 Development (ADV)

5.3.3.1 Informal functional specification (ADV_FSP.1)

ADV_FSP.1.1d The developer shall provide a functional specification.

ADV_FSP.1.1c The functional specification shall describe the TSF and its external interfaces using an informal style.

ADV_FSP.1.2c The functional specification shall be internally consistent.

ADV_FSP.1.3c The functional specification shall describe the purpose and method of use of all external TSF interfaces, providing details of effects, exceptions and error messages, as appropriate.

ADV_FSP.1.4c The functional specification shall completely represent the TSF.

ADV_FSP.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADV_FSP.1.2e The evaluator shall determine that the functional specification is an accurate and complete instantiation of the TOE security functional requirements.

5.3.3.2 Security enforcing high-level design (ADV_HLD.2)

ADV_HLD.2.1D The developer shall provide the high-level design of the TSF.

ADV_HLD.2.1C The presentation of the high-level design shall be informal.

ADV_HLD.2.2C The high-level design shall be internally consistent.

ADV_HLD.2.3C The high-level design shall describe the structure of the TSF in terms of subsystems.

ADV_HLD.2.4C The high-level design shall describe the security functionality provided by each subsystem of the TSF.
ADV_HLD.2.5C The high-level design shall identify any underlying hardware, firmware, and/or software required by the TSF with a presentation of the functions provided by the supporting protection mechanisms implemented in that hardware, firmware, or software.
ADV_HLD.2.6C The high-level design shall identify all interfaces to the subsystems of the TSF.
ADV_HLD.2.7C The high-level design shall identify which of the interfaces to the subsystems of the TSF are externally visible.
ADV_HLD.2.8C The high-level design shall describe the purpose and method of use of all interfaces to the subsystems of the TSF, providing details of effects, exceptions and error messages, as appropriate.
ADV_HLD.2.9C The high-level design shall describe the separation of the TOE into TSP enforcing and other subsystems.
ADV_HLD.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
ADV_HLD.2.2E The evaluator shall determine that the high-level design is an accurate and complete instantiation of the TOE security functional requirements.

5.3.3.3 Informal correspondence demonstration (ADV_RCR.1)

ADV_RCR.1.1d The developer shall provide an analysis of correspondence between all adjacent pairs of TSF representations that are provided.
ADV_RCR.1.1c For each adjacent pair of provided TSF representations, the analysis shall demonstrate that all relevant security functionality of the more abstract TSF representation is correctly and completely refined in the less abstract TSF representation.
ADV_RCR.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.4 Guidance documents (AGD)

5.3.4.1 Administrator guidance (AGD_ADM.1)

AGD_ADM.1.1d The developer shall provide administrator guidance addressed to system administrative personnel.
AGD_ADM.1.1c The administrator guidance shall describe the administrative functions and interfaces available to the administrator of the TOE.
AGD_ADM.1.2c The administrator guidance shall describe how to administer the TOE in a secure manner.
AGD_ADM.1.3c The administrator guidance shall contain warnings about functions and privileges that should be controlled in a secure processing environment.
AGD_ADM.1.4c The administrator guidance shall describe all assumptions regarding user behavior that are relevant to secure operation of the TOE.
AGD_ADM.1.5c The administrator guidance shall describe all security parameters under the control of the administrator, indicating secure values as appropriate.
AGD_ADM.1.6c The administrator guidance shall describe each type of security-relevant event relative to the administrative functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.
AGD_ADM.1.7c The administrator guidance shall be consistent with all other documentation supplied for evaluation.
AGD_ADM.1.8c The administrator guidance shall describe all security requirements for the IT environment that are relevant to the administrator.
AGD_ADM.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.4.2 User guidance (AGD_USR.1)

AGD_USR.1.1d The developer shall provide user guidance.
AGD_USR.1.1c The user guidance shall describe the functions and interfaces available to the non-administrative users of the TOE.

AGD_USR.1.2c The user guidance shall describe the use of user-accessible security functions provided by the TOE.
AGD_USR.1.3c The user guidance shall contain warnings about user-accessible functions and privileges that should be controlled in a secure processing environment.
AGD_USR.1.4c The user guidance shall clearly present all user responsibilities necessary for secure operation of the TOE, including those related to assumptions regarding user behavior found in the statement of TOE security environment.
AGD_USR.1.5c The user guidance shall be consistent with all other documentation supplied for evaluation.
AGD_USR.1.6c The user guidance shall describe all security requirements for the IT environment that are relevant to the user.
AGD_USR.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.5 Life cycle support (ALC)

5.3.5.1 Flaw reporting procedures (ALC_FLR.2)

ALC_FLR.2.1d The developer shall provide flaw remediation procedures addressed to TOE developers.
ALC_FLR.2.2d The developer shall establish a procedure for accepting and acting upon all reports of security flaws and requests for corrections to those flaws.
ALC_FLR.2.3d The developer shall provide flaw remediation guidance addressed to TOE users.
ALC_FLR.2.1c The flaw remediation procedures documentation shall describe the procedures used to track all reported security flaws in each release of the TOE.
ALC_FLR.2.2c The flaw remediation procedures shall require that a description of the nature and effect of each security flaw be provided, as well as the status of finding a correction to that flaw.
ALC_FLR.2.3c The flaw remediation procedures shall require that corrective actions be identified for each of the security flaws.
ALC_FLR.2.4c The flaw remediation procedures documentation shall describe the methods used to provide flaw information, corrections and guidance on corrective actions to TOE users.
ALC_FLR.2.5c The flaw remediation procedures documentation shall describe a means by which the developer receives from TOE users reports and enquiries of suspected security flaws in the TOE.
ALC_FLR.2.6c The procedures for processing reported security flaws shall ensure that any reported flaws are corrected and the correction issued to TOE users.
ALC_FLR.2.7c The procedures for processing reported security flaws shall provide safeguards that any corrections to these security flaws do not introduce any new flaws.
ALC_FLR.2.8c The flaw remediation guidance shall describe a means by which TOE users report to the developer any suspected security flaws in the TOE.
ALC_FLR.2.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.5.2 Identification of security measures (ALC_DVS.1)

ALC_DVS.1.1D The developer shall produce development security documentation.
ALC_DVS.1.1C The development security documentation shall describe all the physical, procedural, personnel, and other security measures that are necessary to protect the confidentiality and integrity of the TOE design and implementation in its development environment.
ALC_DVS.1.2C The development security documentation shall provide evidence that these security measures are followed during the development and maintenance of the TOE.
ALC_DVS.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
ALC_DVS.1.2E The evaluator shall confirm that the security measures are being applied.

5.3.6 Tests (ATE)

5.3.6.1 Analysis of coverage (ATE_COV.2)

ATE_COV.2.1D The developer shall provide an analysis of the test coverage.

ATE_COV.2.1C The analysis of the test coverage shall demonstrate the correspondence between the tests identified in the test documentation and the TSF as described in the functional specification.

ATE_COV.2.2C The analysis of the test coverage shall demonstrate that the correspondence between the TSF as described in the functional specification and the tests identified in the test documentation is complete.

ATE_COV.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.6.2 Testing: high-level design (ATE_DPT.1)

ATE_DPT.1.1D The developer shall provide the analysis of the depth of testing.

ATE_DPT.1.1C The depth analysis shall demonstrate that the tests identified in the test documentation are sufficient to demonstrate that the TSF operates in accordance with its high-level design.

ATE_DPT.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.6.3 Functional testing (ATE_FUN.1)

ATE_FUN.1.1d The developer shall test the TSF and document the results.

ATE_FUN.1.2d The developer shall provide test documentation.

ATE_FUN.1.1c The test documentation shall consist of test plans, test procedure descriptions, expected test results and actual test results.

ATE_FUN.1.2c The test plans shall identify the security functions to be tested and describe the goal of the tests to be performed.

ATE_FUN.1.3c The test procedure descriptions shall identify the tests to be performed and describe the scenarios for testing each security function. These scenarios shall include any ordering dependencies on the results of other tests.

ATE_FUN.1.4c The expected test results shall show the anticipated outputs from a successful execution of the tests.

ATE_FUN.1.5c The test results from the developer execution of the tests shall demonstrate that each tested security function behaved as specified.

ATE_FUN.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.6.4 Independent testing - sample (ATE_IND.2)

ATE_IND.2.1d The developer shall provide the TOE for testing.

ATE_IND.2.1c The TOE shall be suitable for testing.

ATE_IND.2.2c The developer shall provide an equivalent set of resources to those that were used in the developer’s functional testing of the TSF.

ATE_IND.2.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ATE_IND.2.2e The evaluator shall test a subset of the TSF as appropriate to confirm that the TOE operates as specified.

ATE_IND.2.3e The evaluator shall execute a sample of tests in the test documentation to verify the developer test results.

5.3.7 Vulnerability assessment (AVA)

5.3.7.1 Strength of TOE security function evaluation (AVA_SOF.1)

AVA_SOF.1.1d The developer shall perform a strength of TOE security function analysis for each mechanism identified in the ST as having a strength of TOE security function claim.

AVA_SOF.1.1c For each mechanism with a strength of TOE security function claim the strength of TOE security function analysis shall show that it meets or exceeds the minimum strength level defined in the PP/ST.

AVA_SOF.1.2c For each mechanism with a specific strength of TOE security function claim the strength of TOE security function analysis shall show that it meets or exceeds the specific strength of function metric defined in the PP/ST.

AVA_SOF.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AVA_SOF.1.2e The evaluator shall confirm that the strength claims are correct.

5.3.7.2 Developer vulnerability analysis (AVA_VLA.1)

AVA_VLA.1.1d The developer shall perform a vulnerability analysis.

AVA_VLA.1.2d The developer shall provide vulnerability analysis documentation.

AVA_VLA.1.1c The vulnerability analysis documentation shall describe the analysis of the TOE deliverables performed to search for obvious ways in which a user can violate the TSP.

AVA_VLA.1.2c The vulnerability analysis documentation shall describe the disposition of obvious vulnerabilities.

AVA_VLA.1.3c The vulnerability analysis documentation shall show, for all identified vulnerabilities, that the vulnerability cannot be exploited in the intended environment for the TOE.

AVA_VLA.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AVA_VLA.1.2e The evaluator shall conduct penetration testing, building on the developer vulnerability analysis, to ensure obvious vulnerabilities have been addressed.

5.3.7.3 Examination of guidance (AVA_MSU.1)

AVA_MSU.1.1D The developer shall provide guidance documentation.

AVA_MSU.1.1C The guidance documentation shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences and implications for maintaining secure operation.

AVA_MSU.1.2C The guidance documentation shall be complete, clear, consistent and reasonable.

AVA_MSU.1.3C The guidance documentation shall list all assumptions about the intended environment.

AVA_MSU.1.4C The guidance documentation shall list all requirements for external security measures (including external procedural, physical and personnel controls).

AVA_MSU.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AVA_MSU.1.2E The evaluator shall repeat all configuration and installation procedures to confirm that the TOE can be configured and used securely using only the supplied guidance documentation.

AVA_MSU.1.3E The evaluator shall determine that the use of the guidance documentation allows all insecure states to be detected.

6 TOE Summary Specification

This chapter describes the security functions and associated assurance measures.

6.1 TOE Security Functions

6.1.1 Security Audit

The TOE provides its own audit mechanism that generates audit records and displays them in the Manager GUI. When a security-relevant event occurs, the Manager or Gateway involved generates an audit record. To display the audit trail in the Manager GUI, the Manager interacts with the Gateway.

The Manager stores the audit trail in files on its own disk storage. Only the authorized administrator can view the manager audit logs, and they must authenticate in the Manager user interface to do so. Privileged users and non-administrative users can view the event log, and privileged users and non-administrative users that have been assigned the role MTL can view the message traffic log.

To generate the audit trail, the Manager polls its Gateway instances regularly for records of security-relevant events. The Manager writes each event record as a time-stamped entry in one of the three logs that compose the audit trail: the Event Log, the Message Traffic Log, and the Manager Audit Log. Each log contains a particular kind of information:

- Message Traffic Log – Message traffic or information about message traffic passing between consumers and services
- Event Log – Gateway or Manager events that are not the actions of login accountholders
- Manager Audit Log – Actions performed by login accountholders

Each log entry bears a timestamp generated by the operating system. The hardware clock on the server chassis provides a reliable source of time data that the TOE uses to generate timestamps. Generation of this time data is the only security-enforcing behavior the TOE hardware provides. Therefore, all of the functionality of the TOE hardware except its clock functionality is excluded from the TSF. Similarly, generation of timestamps is the only security-enforcing behavior the operating system provides. Therefore, all functionality of the operating system except for its time-stamping functionality is excluded from the TSF.

The Message Traffic Log contains the following additional information on events that are collected: a timestamp, the IP address of the particular Gateway machine that recorded it, the message type, the handler that recorded it, the authenticator that matched the request, and details of the actual request and response.

The TOE relies on the operating system and server chassis hardware clock to provide a reliable source of time data for its use in the generation of time stamps. Because the appliance uses the server chassis "off the shelf" with no modifications, the evaluated configuration relies on this hardware clock to provide the reliable time stamp for the TOE.

The Cisco modifications to the base Linux operating system consist of package-level reductions, as specified in the *Cisco Systems ACE XML Gateway and Manager Version 5.0.3 Configuration Management Procedures* document. The remaining packages that compose the embedded OS are "off the shelf" and unmodified. Therefore the evaluated configuration relies on this timestamp-generation functionality of the operating system to provide the reliable time stamp for the TOE.

In addition to describing the date and time of the event, each log entry specifies the type of event, subject identity, and the outcome (success or failure) of the event. Events in the separate logs can be correlated by subject identity in order to compose the audit trail.

The auditable events include:

- Start-up and shutdown of the audit function (audit record generated by Gateway or Manager involved) **Note:** The audit function starts automatically when the TOE starts up and the audit function shuts down automatically when the TOE shuts down. The TOE does not provide the capability to disable audit functionality. The audit function always runs when the TOE runs.

- Decisions to permit requested information flows (audit record generated by Gateway)
- The reaching of the threshold for unsuccessful authentication attempts and the actions taken (audit record generated by Manager)
- Successful and unsuccessful use of the authentication mechanism (audit record generated by Manager)
- Successful and unsuccessful use of the user identification mechanism, including the user identity provided (audit record generated by Manager)
- Successful or unsuccessful assignment of security attributes, including administrative roles (audit record generated by Manager)
- Successful or unsuccessful revocation of security attributes, including administrative roles (audit record generated by Manager)
- Use of the management functions: (audit records generated by Manager)
  - Manage Web services
  - Manage users
  - Manage the audit trail
- Message traffic (audit record generated by Gateway). The information in the audit record includes time of arrival at the Gateway; hostname or address requested; a summary of any errors the message produced; the message's type; the sender's IP address and (if available) authenticator name; the name of the handler that matched the message; the service to which the Gateway routed the message; attributes of the request/response message pair associated with the log entry; and a list of all Event Log entries associated with the message.

Only authorized administrators and users who have been assigned appropriate security roles can view the TOE audit trail in the Web-based user interface to the Manager.

The Security Audit function is designed to satisfy the following security functional requirements:

- FAU_GEN.1: The TOE generates audit events for startup and shutdown of the audit function, all auditable events for the minimum level of audit and for all attempted or successful modifications of TSF data. **Note:** The audit function starts automatically when the TOE starts up and the audit function shuts down automatically when the TOE shuts down. The TOE does not provide the capability to disable audit functionality. The audit function always runs when the TOE runs.
- FAU_SAR.1: The TOE provides Manager GUI interfaces that can be used to read audit trail files.
- FAU_STG.1: The TOE restricts access to the audit trail to authorized administrators and users who have been assigned appropriate security roles by requiring successful identification and authentication as a prerequisite to viewing or querying the audit trail. The evaluated configuration of the TOE provides no means of editing or deleting the audit trail or modifying the individual records that compose it.

6.1.2 User Data Protection

The TOE implements an information flow policy (WEB SERVICES SFP) for Web service message traffic based on:

- Subject Security Attribute
  - Consumer network addresses (presumed)
- Information Security Attributes
  - Consumer user identity (user name and password)

  - Consumer credential attributes (SSL Client Certificate Distinguished Name Parameter and X.509 Certificate)
  - Port on which a handler accepts messages
  - Local endpoint descriptor a handler exposes to consumers
  - Version of SOAP to use for transactions
  - Requested URI
  - Requested RPC method name
  - Requested SOAP Action

All non-administrative data from the untrusted network is subject to the Web services SFP. The TOE can proxy these message types using HTTP(S) network protocols.

The TOE offers a choice of two built-in message-processing engines. The evaluated configuration allows use of the standard engine only. The evaluated configuration excludes use of the optional "Reactor" engine. Authorized administrators can disable Reactor on a per-port basis (optionally, restricted to requests to a specific hostname or IP address) by checkmarking the "Always Use Flex Path" option on the **Edit Port** page in the Manager GUI; in the evaluated configuration, this box must be checkmarked.

Consumers are Web service clients that attempt to connect to Web services. An authenticator specifies the data needed to positively identify a particular consumer. The evaluated configuration excludes the use of authenticators provisioned as "Public." Such provisioning allows access to an authorization group's handlers without requiring the message sender to provide any authentication data.

An authenticator filters messages (FDP) on the presence of authentication data, but it does not use this data to create a user session with the ACE XML Manager or Gateway. Authenticators are not user accounts and a consumer who sends a message to service the ACE XML Gateway protects has not logged into the TOE. For more information, see Section 6.1.4, "Security Management."

Authorization groups are collections of authenticators with assigned permissions. For example, when a consumer tries to connect to a service protected by the TOE, it is rejected unless the authenticator that accepts the message belongs to the same authorization group as a handler that can route messages to the specified service.

The TOE stores subject and information security attributes in its configuration files. The consumer's network address, consumer's username/ password, and Web service information are used by the WEB SERVICES SFP mechanism to make decisions to permit requested information flows.

Only an authorized administrator can configure the TOE for use with a Web service and grant permission for a consumer to use that service. Services are defined as collections of network endpoints, or ports. The abstract definition of endpoints and messages is separated from their concrete network deployment or data format bindings. This allows the reuse of abstract definitions: messages, which are abstract descriptions of the data being exchanged, and port types which are abstract collections of operations. The concrete protocol and data format specifications for a particular port type constitute a reusable binding. A port is defined by associating a network address with a reusable binding, and a collection of ports define a service. Services are defined using the following types of information:

- Types – a container for data type definitions using some type system (such as XSD).
- Message – an abstract, typed definition of the data being communicated.
- Operation – an abstract description of an action supported by the service.
- Port Type –an abstract set of operations supported by one or more endpoints.
- Binding – a concrete protocol and data format specification for a particular port type.
- Port – a single endpoint defined as a combination of a binding and a network address.
- Service – a collection of related endpoints.

In addition, the TOE provides the ability for authorized administrators to specify the following information for a Web service in order to support the WEB SERVICES SFP:

- Access control settings – an authorization group associated with a handler specifies the set of authenticators that can pass messages to the handler. Routes specify the services with which the handler can exchange message data. Handlers and service descriptors specify available URIs.
- Encryption requirements – imposed by handlers and service descriptors, these requirements are associated with individual messages and network connections
- Digital signature requirements – imposed by handlers and service descriptors, these requirements are associated with individual messages
- Handling of SOAP attachments – imposed by handlers and service descriptors, these requirements are associated with blocks of data included in a SOAP message as a MIME or DIME attachment

Consumer access to a service the Gateway protects is predicated on the incoming request message meeting all criteria the Web services SFP defines for valid messages from a particular sender; subsequently, the consumer's request message must meet all additional criteria the Web services SFP defines.

To define the criteria against which the Web Services SFP initially tests messages incoming to the Gateway, authorized administrators create and edit authenticators in the Manager user interface. Each **authenticator** specifies the tests a message must pass in order for the Web Services SFP to accept the message as having originated from a known sender. An individual test is called a **credential**, and the portion of the Web Services SFP that uses credentials to filter incoming messages is known as the **credentialing mechanism**.

The Web Services SFP accepts for further processing only those messages that provide all of the credentials a defined authenticator specifies. If an incoming message does not meet all requirements of a defined authenticator, the Web Services SFP does not process the message further and does not pass the message to the trusted network.

For use by the Web Services SFP, an Authenticator supports the following Web service authentication mechanisms that may be available in the IT environment:

- HTTP Basic authentication mechanism
- X.509 certificate authentication mechanism
- SAML authentication mechanism
- LDAP authentication mechanism

For consumer credentials based on username/password mechanisms, the Gateway stores encrypted passwords in configuration files on its local disk storage. The Gateway can also use SSL to protect connections from consumers to Web services.

Acceptance of an incoming request message by an authenticator in the Web Services SFP means only that the Gateway has not blocked the incoming message based on characteristics of the sender—it does not guarantee that the request message will reach the Web service the TOE protects. Upon accepting the request message from the authenticated consumer, the Gateway processes the message further, according to the Web services SFP an authorized administrator has defined for messages of that type from that consumer for the specified destination; unless the message meets every criterion imposed by the authorization group, handler, route, service descriptor and other elements of the Web Services SFP (such as global content-screening rules), the Gateway does not pass the request message to the destination service, and the audit trail logs the failed request. Responses from the destination service to the consumer are also processed similarly: unless the response message meets all criteria the Web services SFP imposes, the response is blocked and the failure logged.

In the evaluated configuration, the audit trail logs startup and shutdown of the audit function, all auditable events for the minimum level of audit, message traffic or information about message traffic passing between consumers and services, and all attempted or successful modifications of TSF data. Optionally, authorized administrators can configure the Event Log and Message Log to record additional information about events or messages.

Note: The audit function starts automatically when the TOE starts up and the audit function shuts down automatically when the TOE shuts down. The TOE does not provide the capability to disable audit functionality. The audit function always runs when the TOE runs.

After meeting all requirements imposed by a defined authenticator and any global content-screening rules or denial-of-service thresholds that are in effect, a message must meet all criteria imposed by an authorization group, handler, route, and service descriptor. Not only request messages but also responses from a protected service must meet these criteria (except that response messages are not tested against an authorization group.) If all of these criteria are not met, the message is not allowed to pass through the Gateway.

The remainder of this section refers to a message that meets all requirements imposed by a policy object (such as an authenticator, handler, route, or service descriptor) as one that “matches” the policy object. For example, a message that meets all criteria imposed by a handler is said to **match** that handler.

A **handler** manages communications between the Gateway and the sender of a request message. A **request message** is a message incoming to the Gateway that requests access to a service that the Gateway protects. The Web Services SFP may define multiple handlers. Each handler exposes a network endpoint on the untrusted network and defines specific criteria that a message must meet in order to be accepted by that endpoint. Criteria that a handler specifies always include the URI/port to which the message must be addressed and the Web services protocol that the message must use. Optionally, the handler may impose additional requirements, such as those which define the structure of an acceptable message, digital signature requirements, required or forbidden attachments, and so on. In short, the handler defines the first of several sets of destination-specific requirements that an incoming request message must meet. For example, a handler accepts only messages addressed to the URI on which it listens for incoming traffic, and it ignores all other requests. Within transport-specific limitations, the criteria a handler imposes can be as restrictive or permissive as the authorized administrator of the TOE requires; for example, the handler can be configured to accept messages for a wide range of URIs and send them to a variety of destinations, or it can be configured to reject all messages that do not meet a very restrictive set of criteria.

Simply matching a handler and an authenticator does not provide access to the requested service. The matched handler and authenticator must both be members of the same authorization group. An **authorization group** specifies the set of authenticators from which a handler accepts requests. A handler that is assigned to an authorization group is said to be **provisioned** to that authorization group; a handler that is not provisioned cannot accept any incoming messages. A handler that is provisioned to the factory-configured Public authorization group imposes no authenticator-based requirements on incoming messages; essentially, it bypasses the consumer credentialing mechanism of the Web Services SFP. The evaluated configuration of the TOE expressly forbids the provisioning of any handler to the Public authorization group.

A message that matches a provisioned handler must then match a route associated with the handler. A **route** passes the message to a **service descriptor**, which manages the interaction of the TOE with a Web service. Multiple routes may be assigned to a handler, allowing it to send messages to different service destinations according to criteria that the matched route imposes. For example, a handler might accept messages that use any version of SOAP and then use route-based criteria to send the messages to various destinations according to the version of SOAP that the message actually uses. Like the other policy objects mentioned thus far, each route accepts only messages that match all of the criteria it defines. Route-based criteria are entirely administrator-defined and may examine virtually any aspect of the incoming message for any value or set of values the authorized administrator specifies, including custom-defined headers and message-body content.

The matched route sends the message to a specific service descriptor. As noted earlier, the service descriptor manages the interaction of the Gateway with a Web service it protects; thus, the service descriptor specifies the URI on which the service accepts request messages coming from the Gateway on behalf of authenticated consumers. This URI need not be identical to the one the handler specifies. The service descriptor also specifies the IP address from which authorized communications from the protected service originate. Optionally, the service descriptor can specify additional criteria that the Gateway uses to authenticate the protected service, such as requiring it to present an X.509 certificate signed by a trusted CA and requiring use of a protected connection such as HTTPS.

As specified by an authorized administrator configuring the Web Services SFP in the appropriate editor of the Manager GUI, the handler or the service descriptor (or both) can impose additional requirements on the content or structure of request or response messages to ensure their validity. These requirements may include SOAP version requirements, SOAP RPC requirements, digital signature requirements, encryption requirements, SAML assertion requirements, attachment validation or handling requirements, and DTD-or-schema-based validation at administrator-specified levels of rigor.

Note that both the handler and the service descriptor specify acceptable URIs and other pathname-based requirements for message traffic directed at the Gateway, and that a specific route must connect a specific handler to a specific service descriptor. Therefore, the combination of handler, route, and service descriptor mutually enforce restrictions on acceptable values for hostname, port, pathname, SOAP Action/RPC, and any other parameters that may be expressed within a URI.

The User data protection function is designed to satisfy the following security functional requirements:

- FDP_IFC.1: The TOE enforces the WEB SERVICES SFP on HTTP(S) destination service traffic sent through the TOE from one subject to another (consumer to Web service and vice-versa).
- FDP_IFF.1: The TOE enforces the WEB SERVICES SFP, using Authenticators to verify the user and group identity of a consumer of a service, using handlers to validate incoming messages, using routes to pass accepted message to service descriptors, and using service descriptors to manage traffic with SOAP or HTTP(S) destination services.

6.1.3 Identification and Authentication

The TOE provides its own identification and authentication mechanism for user logins and the TOE provides no access to TSFs until the user identifies and authenticates successfully. Administrative access to the Gateway using the Manager requires successful authentication to a local user account on the Manager.

A **local user account** is a login account that utilizes the Manager's local authentication mechanism. This account specifies the username and password a user must supply to the graphical user interface of the Manager in order to log into the Manager successfully. In the evaluated configuration, only the factory-configured "administrator" account can create a login account and initialize or set the values of security attributes associated with the account. These security attributes specify the username, password, roles, and enabled/disabled status of the account. A user or an authorized administrator can modify and reset its own account's password. The factory-configured “administrator” account specifies that a user account is an authorized administrator by assigning administrative roles to the account, as described in relation to the Security Management function (below).

The Manager appliance stores each local account username and password on its local disk as part of TOE configuration data. Using the FIPS 140-2 validated cryptomodule supplied by the IT environment, it stores the password in encrypted (hashed) format only. In addition to the user's identifiers and password, any groups and roles assigned to the user are also stored as part of TOE configuration data. Note that groups are used to simplify access control management. Although the “sub-policy” feature of the Manager is intended to implement group-level access controls on Manager account access to policy objects, the evaluated configuration requires use of the default “Shared” sub-policy at all times and forbids the creation or use of other sub-policies. By default, all users belong to the group that has access to the Shared sub-policy. Effectively, all Manager user accounts and all policy objects belong to a single group in the evaluated configuration.

To log on to the TOE, the holder of a user account provides the correct login name and password to the Manager's administrative graphical user interface (the Manger GUI). To protect TSF data (including authentication data) from disclosure, an SSL connection between the Manager GUI on the accountholder's Web browser and the Manager appliance encrypts TSF data (again using the IT cryptomodule) and carries it on a trusted channel. In addition to encrypting the connection, keys, and TSF data, the SSL protocol incorporates multiple mechanisms that ensure the authenticity of the sender and the integrity of the data, including the use of message authentication codes (MACs) that prevent tampering with encrypted data.

Once within the boundaries of the TOE, the login data is authenticated by the TOE only, without use of any external services; hence, the name “local” authentication mechanism. Although the Manager supplies alternative mechanisms for authenticating logins, only “Standard ACE XML Manager Passwords” authentication mode (aka "local authentication mode") is the evaluated configuration.

As a result of a successful login, a subject is created on behalf of the client, and the audit trail logs the successful authentication. If authentication is not successful, the TOE makes no TSF-mediated functions available and the audit trail logs the failed request. After an administratively-configurable number of failed authentication attempts (three attempts by default), the Manager disables the account that failed. In the evaluated configuration, only the factory-configured "administrator" account may re-enable the disabled account. Should the factory-configured "administrator" account be disabled due to authentication failures, it may be re-enabled by restarting the Manager application; to do so requires either physical access to the Manager appliance or terminal access as the trusted "root" user in a UNIX terminal session. Without physical access to the administrative terminal connected directly to the Manager appliance, root user access to the Manager appliance is available only over a password-protected, encrypted, "secure shell" (ssh) UNIX terminal session. The default setting of three consecutive failed attempts causing the account to become disabled is given as the recommended setting in the Administrative Guidance. Warnings are also given against setting the value too high, or to zero, which disables this feature.

The Identification and authentication function is designed to satisfy the following security functional requirements:

- FIA_AFL.1: The TOE disables user or administrator accounts after an administrator configurable setting of failed login attempts to the Manager. The default is three failed login attempts and is the recommended setting for the evaluated configuration.
- FIA_ATD.1: The TOE maintains user identities, authentication data for supported authentication mechanisms, and role information.
- FIA_UAU.2: The TOE offers no TSF-mediated functions until the user is authenticated.
- FIA_UID.2: The TOE offers no TSF-mediated functions until the user is identified.
- FPT_ITT.1: The TOE protects TSF data from disclosure while in transit by using an encrypted, private channel to carry TSF data between the Web browser of a user and the Manager, and between the Manager and the Gateway.

6.1.4 Security Management

The Manager component of the TOE provides a GUI to administer the Gateway component of the TOE. The Manager can be used for creating, managing, and deploying WEB SERVICES SFP security policies, for managing users, and for monitoring the Gateway, including its audit trail.

Authorized administrators are users that log into the Manager to perform administrative actions. All other TOE users are non-administrative

There are two types of authorized administrators:

1. Factory-configured `administrator` - One, specially-privileged ACE XML Manager user account that has the ConsoleAdmin role, which grants permission to view, configure, and control every aspect of ACE XML Manager and Gateway operation, including the ability to perform any task that any other account can perform. This account, named `administrator`, is created at the Cisco Systems factory and cannot be renamed or deleted. It is the only account that can view the Manager Audit Log, create and manage accounts, and assign roles.
2. User with assigned roles – Individual user accounts on the Manager that have been assigned one or more of the following roles: "Operations", "Access Control", "Message Traffic Log (MTL)", or "Routing". The privileges vary based on the assigned role: Access Control grants access to edit authenticators; Routing grants access to edit handlers, edit service descriptors, edit routes, and edit message transformers; Operations grants access to view and deploy policies; and MTL grants access to query the Message Traffic Log. All of these user accounts can query the Event Log, view the services directory, and view the Web Services SFP.

   Note: The evaluated configuration expressly forbids assignment of the ConsoleAdmin role to any account other than the factory-configured `administrator` account. In the evaluated configuration, only the factory-configured `administrator` account has this role.

Any authorized administrator or user account can modify or reset its own password. However, the factory-configured `administrator` account is the only account that can edit other user security attributes and edit other user accounts, including the user account's authentication data, account type, roles assigned, and enabled/disabled status. In particular, the factory-configured `administrator` account is the only one that can initialize or set the user authentication data of another user's account.

The **user** category comprises view-only accounts (External Developer and Policy View) on the Manager.

Policy View accounts can view but not edit the working policy, which resides on the Manager. (The Web Services SFP is the active policy, which runs on the Gateway.) All Policy View users can query the Event Log. An individual Policy View user can query the Message Traffic Log only if the Message Traffic Log role is assigned to that user's account; otherwise, the user cannot even view this log. A Policy View user has no access to any other TSFs reserved for authorized administrators.

An External Developer account can view the directory of services that the Gateway publishes, but has no access to any other TSFs reserved for authorized administrators. The directory of services displays a read-only subset of information security attributes of the Web Services SFP (not the working policy) that includes the names of handlers and their authorization groups, the externally-resolvable hostname each handler publishes, and the request and response message specification that each handler defines. For more information, see descriptions of the service directory, handlers, authorization groups, and message specifications in the *Using the Cisco Systems ACE XML Gateway and Manager Version 5.0.3* user guidance document.

Consumers are not users on the TOE; rather, they are users of proxying services that the TOE provides. Consumers never interact directly with the TOE and may not even know that it exists. A message sender who supplies data to an authenticator's consumer credentialing mechanism fulfills one prerequisite to sending a message to a Web service that the Gateway protects—the message sender has not logged on to the TOE and is not a user on the TOE.

To clarify this distinction, it is important to understand that an authenticator is a message-traffic filter on the Gateway, not a user account on the Manager. An authenticator specifies one set of criteria on which the Gateway screens incoming messages. When the headers of a message present valid values to an authenticator, it simply means that the Gateway did not reject the message based on the criteria the authenticator defines. It does not mean that the message sender has logged in to the Manager. If a message matches an authenticator, the Gateway may still reject the message based on other criteria the Web Services SFP specifies. Thus, an entity (a consumer) that supplies required attribute values to an authenticator is not an administrative user, has not logged onto the TOE, does not have a user account on the TOE, cannot be assigned administrative roles, cannot perform administrative tasks and is not even considered a user "on" the TOE so much as a user "of" services the TOE provides.

The use of a UNIX shell or remote terminal in the IT environment is excluded from the evaluated configuration for all but the following uses by the trusted root user:

- Installation, Generation and Startup of the TOE
- Re-enabling the factory-configured administrator account after it has been disabled automatically due to failed login attempts.
- Backup of log files, X.509 certificates, archived security policies, and other file-based resources.
- Moving log files to offline storage in order to maintain adequate free space on local storage for the normal operation of the TOE.

When the root user completes these activities and the TOE is in the secure default state, the root user must exit all terminal shells. Except for the activities noted above, users must perform all security management functions from within the Manager GUI.

To configure the TOE for use with a Web service, an authorized administrator uses the Manager GUI to perform the following steps:

1. Create a WEB SERVICES SFP policy to handle transmitted messages. The WEB SERVICES SFP policy is composed of the following policy objects that provide restrictive default values and allow an authorized administrator to specify alternative initial values:
    a. Create Authenticators, which verify the identities of consumers attempting to connect to services the Gateway protects.
    b. Create Handlers, which listen for message traffic on specific ports and URIs.

   c. Enable access for authorized consumers on ports and URIs by grouping Authenticators with Handlers in Authorization Groups.
   d. Create Service Descriptors, which manage traffic with SOAP or HTTP(S) destination services.
   e. Create Routes, which pass messages from handlers to service descriptors according to the values of specified message attributes.
   f. Configure Denial-of-Service protection and default Content Filtering settings.
2. Perform an automated review of each policy before loading it onto the Gateway.
3. Load completed policies onto Gateway machines for enforcement.

Loading a completed policy onto a Gateway for enforcement is known as **deploying** the policy to the Gateway. The Manager's monitoring and diagnostic tools can be used after a policy has been deployed in order to ensure that the Gateway is configured correctly and is performing as expected. Note that the Manager includes the ability to perform self tests, which are described in the Protection of the TSF security function description section below.

The Security management function is designed to satisfy the following security functional requirements:

- FMT_MOF.1: The TOE restricts the ability to assign roles required to specify Web services policy to the ConsoleAdmin authorized administrator.
- FMT_MSA.1: The TOE restricts the ability to modify or delete subject security attributes, information security attributes, and user security attributes to authorized administrators having sufficient roles. The TOE restricts the ability to view subject security attributes to users that have the Policy View role and authorized administrators. The TOE restricts the ability to view information security attributes to users that have the External Developer role, users that have the Policy View role, and authorized administrators.
- FMT_MSA.3: The TOE provides restrictive default values for security attributes used to enforce the WEB SERVICE SFP. The TOE also allows authorized administrators to specify alternative initial values.
- FMT_MTD.1: The TOE restricts the ability to initialize and set user authentication data to authorized administrators having sufficient roles. The TOE restricts the ability to modify and reset an account's own password to authorized administrators and users. The TOE restricts the ability to view or query audit records to authorized administrators having sufficient roles, with the exception that users who have the Policy View role can query the Event Log and users who have the MTL role can query the Message Traffic Log.
- FMT_REV.1: The TOE restricts the ability to revoke user security attributes to authorized administrators by restricting access to user security attribute editors in the Manager GUI to authorized administrators. The enforcement of changes to security attributes takes place immediately upon completion of the revocation operation.
- FMT_SMF.1: The TOE provides authorized administrators with the ability to manage Web services, to manage users, and to manage the audit trail using the Manager.
- FMT_SMR.1: There are two categories of administrators on the TOE: The first is the default factory administrator account that has all of the administrative privileges for the TOE, and includes the privileges of all of the other roles. The second category of administrators is a user that has been assigned one or more system-defined roles: "Operations", "Access Control", "MTL", or "Routing". These are considered to be "authorized administrators". There is a third category of users, those with view-only privileges: External Developer, and Policy View, which are considered simply "users." The single factory-configured `administrator` account always has all roles, cannot be modified or deleted, and is also considered an "authorized administrator" in terms of the security functional requirements in this ST.

6.1.5 Protection of the TSF

Upon startup, the TOE enters a restrictive default state in which no users are logged into the Manager and no traffic passes through the Gateway. Because the TOE cannot be bypassed, and the Gateway accepts no user logins, this default state is secure: the Gateway enforces the current Web Services SFP independently of the Manager, the

Gateway accepts changes to the current Web Services SFP only from its assigned Manager, and the user interface to the Manager provides no access to TSFs until the user identifies and authenticates successfully.

Upon startup, the TOE attempts to resume its last mode of operation (running/stopped). Assuming the TOE has not been stopped by its administrator, each Gateway instance attempts to load and run the current Web Services SFP; similarly, each Manager instance attempts to load the current working policy and resume running with no users logged in. If the Gateway cannot load and run the current Web Services SFP, it loads and runs the default policy, which allows no traffic to pass through the Gateway.

The Manager ensures that access to TOE configuration data is restricted to authorized administrators by controlling access to its GUI. Similarly, the Gateway ensures that access to Web services using the TOE is restricted to authorized consumers by controlling access to its network interfaces. Access and use of the command line interface (CLI) is also limited. The only account that has login capabilities is the root account, all other accounts do not have login prompts or no login options. The Guidance document specifies the functions that are performed via the CLI.

In order to define Web Services SFPs or administer the Gateway, authorized administrators must authenticate to the Manager via the local account (Standard) authentication mechanism (Identification and Authorization subsystem). The TOE uses its SSL mechanism to protect authentication data and other TSF data when it is transmitted between the authorized administrator's Web browser and the Manager. The TOE also uses SSL to protect TSF data transmitted between the Manager and the Gateway; as noted earlier, the SSL protocol incorporates multiple mechanisms that ensure the authenticity of the sender and the integrity of the data, such as the use of message authentication codes (MACs) that prevent tampering with data, including authentication data and other TSF data.

After authenticating to the Manager, authorized administrators use the Manager's graphical user interface to define rules that permit information flows between consumers and services. Collectively, these rules compose the “working policy.” The Manager user interfaces provide restrictive default values that authorized administrators can modify; when possible, the Manager GUI restricts the values specified by authorized administrators to valid ones.

The “working policy” resides on the Manager appliance. The “active policy” is a policy that is running on a Gateway appliance, which enforces the active policy on traffic that flows between the trusted and untrusted networks.

Only one policy may be the active policy at any time; to clarify, the active policy is the Web Services SFP. Other versions of this policy may exist on the Manager for backup or archival purposes, but only the policy actually running on the Gateway is the Web Services SFP.

To enforce a policy as the active policy (Web Services SFP), an authorized administrator uses the Manager GUI to compile the working policy into a proprietary executable format and transmit it in encrypted form from the Manager to the Gateway over a secure, private channel. When the Gateway receives the encoded binary, it loads and runs the new policy immediately. If the Gateway encounters errors while attempting to load and run the new policy, it does not load and run the new policy, but logs the errors and continues to run the current policy.

The process of loading and running the compiled policy on each Gateway instance is known as “deploying” the policy to the Gateway instance. In the evaluated configuration the “approval-based deployment” feature of the ACE XML Manager and Gateway is not to be enabled. The “standard” deployment mode is to be used in the evaluated configuration. When enabled, an approval-based deployment only allows privileged users (with permissions to edit or deploy policies) to view the section of the GUI, and the changed policy must be approved by the administrator. A standard deployment allows users with the proper permissions to deploy policies without formal approval.

Each Gateway instance accepts a new Web Services SFP from only one assigned Manager instance. Manager and Gateway instances must authenticate to each other in a bilateral X.509 certificate exchange on a trusted channel before the Gateway accepts a new active policy from its Manager, or before any other communications between the Gateway and its Manager.

To ensure that the strength of cryptographic operations is sufficient to defeat an attacker having the identified resources and motivation, the TOE requires the IT environment to supply a FIPS 140-2 validated cryptographic module that it uses as FCS_COP.1 requires. The evaluated configuration uses an "off-the-shelf" nCipher 4000 PCI hardware cryptomodule with no modifications, so the ST treats it as supplied by the IT environment even though it resides on the chassis of the TOE. The nCipher card is initialized in “strict FIPS mode” and used exactly as specified by the validation testing; therefore, the dependencies of key generation, key destruction, and secure key values are satisfied by this module's validation as FIPS PUB 140-2 compliant. For additional information, see Note 1 in Section 8.5, ”Requirement Dependency Rationale.”

The restrictive default state, identification and authentication requirements, encryption of TSF data (including authentication data), and use of a secure private channel for TSF data combine to ensure the protection of the TOE and TSF data from disclosure and misuse.

The TOE ensures the integrity and correct operation of the TSF in the following ways:

- The initial state of all Gateway instances is deny-by-default: no traffic in or out. This state may be changed only by loading and running a compiled policy (the Web Services SFP) that resides on local storage on the Gateway. The compiled policy can be loaded on the Gateway only by an authorized administrator who uses the Manager GUI to define and deploy the policy. The Web Services SFP defines information flow rules which specify the criteria network traffic must meet to pass through the Gateway. The Gateway permits only those information flows which meet all criteria the active Web Services SFP imposes. Each information flow rule is composed of a collection of independent code objects, all of which must load and execute successfully to allow the information flow that the particular rule defines; if this condition is not met, the flow is not permitted. Therefore, inability to load or execute an individual rule does not compromise the secure state of the TOE.

- Before transmitting a Web Services SFP policy to the Gateway, the Manager attempts to ensure the policy’s validity by performing the following operations:

    - First, the Manager conducts an automated test of the Web Services SFP and displays to the authorized administrator a list known and potential problems for correction or explicit authorization to attempt to run 'as is.' This test identifies configuration problems that might interfere with successful information flow, as well as potential security problems.
    - Next, the Manager displays to the authorized administrator an itemized list of every change that deploying the new policy will introduce to the current Web Services SFP, allowing the administrator to reject unwanted changes individually before continuing the deployment process.
    - The Manager then compiles the approved policy into a proprietary binary executable format; only policies that compile without errors may be transmitted to the Gateway.
    - For additional protection against accidental deployment, the final step in the deployment process in the Manager user interface requires the authorized administrator to authorize transmission of the new policy to each Gateway instance explicitly. The evaluated configuration requires all Gateways to run the same version of the policy.

- Once the new policy arrives at the Gateway, the Gateway attempts to load the policy and run it as the Web Services SFP. If the Gateway encounters errors loading or running the new policy, it logs the errors, does not run the new policy, and continues to run the current policy.

Once the Manager and Gateway are up and running, the mechanisms that provide maintenance modes and auditing continue to operate. If configured to do so, the TOE can send email notifications of security-relevant events to email addresses specified by authorized administrators. For example, an authorized administrator can configure a handler to send email when it rejects a message or a specified number of them, or when any of a wide variety of administrator-configurable exceptions occurs.

The TOE provides the ability for authorized administrators to check the configuration of individual Web Services SFP rules and the underlying network by sending test messages to a configured handler/route/service descriptor/Web service message-processing path. The TOE can also protect against denial of service attacks designed to render Web services inaccessible to their legitimate users; when the rate of requests from a particular IP address exceeds a policy-specified threshold, the TOE blocks requests from that address for a policy-specified amount of time.

The Protection of the TSF function is designed to satisfy the following security functional requirements:

- FPT_ITT.1: The TOE uses a FIPS-certified cryptographic module in the IT environment for key generation and protection. The FIPS certificates for the nCipher cryptographic module (nCipher 4000 PCI) are 669 and 670.

  The IT environment cryptographic module is used primarily to generate and protect the private RSA keys. Storing keys in this hardware key store defeats an attacker's chances of stealing them, even if the attacker gains full root ownership of the device.

  The ACE XML Manager always initiates the communication with the ACE XML Gateway, at which time the RSA keys are used only to verify the cryptographic signature over certificates and establish a session key for the Manager and Gateway communication. Once the session key has been established, OpenSSL is used as the SSL implementation to protect the transmitted data. It should be noted that OpenSSL is used throughout the lifetime of the SSL session, not just after the session key has been established. The software library handles all aspects of SSL, except RSA.

  The version of OpenSSL is openssl-0.9.7a-33.21.Reactivity.2.i686.rpm. This version of OpenSSL is not FIPS certified, though all of the applicable patches have been installed to mitigate the known vulnerabilities. To further secure the TOE and the communication between the Gateway and Manager, the TOE is located in a secure and physically protected environment and the network for the Manager and Gateway is a dedicated secure network.

- FPT_RVM.1: The WEB SERVICES SFP cannot be bypassed by consumers. Similarly, both Gateway and Manager interfaces are restricted to authorized administrators and users.

- FPT_STM.1: The operating system provides a reliable timestamp.

6.2 TOE Security Assurance Measures

6.2.1 Configuration management

The configuration management measures applied by Cisco Systems ensure that configuration items are uniquely identified, and that documented procedures are used to control and track changes that are made to the TOE. Cisco Systems performs configuration management on the TOE implementation representation, design, tests, user and administrator guidance, and the CM documentation. TOE is labeled with its reference. A CM Plan is included that describes how the TOE and configuration items are uniquely identified.

These activities are documented in:

- *Cisco Systems ACE XML Gateway and Manager Version 5.0.3 Configuration Management Procedures*

The Configuration management assurance measure satisfies the following EAL 3 augmented with ALC_FLR.2 assurance requirements:

- ACM_CAP.3
- ACM_SCP.1

6.2.2 Delivery and operation

Cisco Systems provides delivery documentation and procedures to identify the TOE, secure the TOE during delivery, and provide necessary installation and generation instructions. Cisco Systems' delivery procedures describe all applicable procedures to be used to prevent inappropriate access to the TOE. Cisco Systems also provides documentation that describes the steps necessary to install the Cisco Systems ACE XML Gateway and Manager Version 5.0.3 in accordance with the evaluated configuration.

These activities are documented in:

- *Cisco Systems ACE XML Gateway and Manager Version 5.0.3 Delivery and Operation Procedures*

- “Creating the Common Criteria Evaluated Configuration” chapter of *Using the Cisco Systems ACE XML Gateway and Manager Version 5.0.3*

The Delivery and operation assurance measure satisfies the following EAL 3 augmented with ALC_FLR.2 assurance requirements:

- ADO_DEL.1
- ADO_IGS.1

6.2.3 Development

Cisco Systems documents the design of the TOE. These documents serve to describe the security functions of the TOE, its interfaces both external and between subsystems, the architecture of the TOE (in terms of subsystems), and correspondence between the available design abstractions (including the ST).

These activities are documented in:

- *Cisco Systems ACE XML Gateway and Manager Version 5.0.3 High Level Design Document*
- *Cisco Systems ACE XML Gateway and Manager Version 5.0.3 Functional Specification*

The Development assurance measure satisfies the following EAL 3 augmented with ALC_FLR.2 assurance requirements:

- ADV_FSP.1
- ADV_HLD.1
- ADV_RCR.1

6.2.4 Guidance documents

Cisco Systems provides administrator and user guidance on how to utilize the TOE security functions and warnings to administrators and users about actions that can compromise the security of the TOE.

These activities are documented in:

- *Using the Cisco Systems ACE XML Gateway and Manager Version 5.0.3*

The Guidance documents assurance measure satisfies the following EAL 3 augmented with ALC_FLR.2 assurance requirements:

- AGD_ADM.1
- AGD_USR.1
- AVA_MSU.1

6.2.5 Life cycle support

Cisco Systems employs a process in which security flaws discovered by customers and Cisco Systems are tracked and corrected by the developer. Cisco Systems bug reconciliation process provides assurance that the TOE is maintained and flaws are corrected in the TOE. Physical, procedural, personnel, and other security measures that are necessary to protect the confidentiality and integrity of the TOE design and implementation in its development environment are described in life cycle documentation.

These activities are documented in:

- *Cisco Systems ACE XML Gateway and Manager Version 5.0.3 Flaw Remediation Procedures*

The Life cycle support assurance measure satisfies the following EAL 3 augmented with ALC_FLR.2 assurance requirements:

- ALC_DVS.1
- ALC_FLR.2

6.2.6 Tests

The test documents describe the overall test plan, testing procedures, and the tests themselves, including expected and actual results. In addition, these documents describe how the functional specification has been appropriately tested.

These activities are documented in:

- *Cisco Systems ACE XML Gateway and Manager Common Criteria Specific Test Plan*
- *Cisco Systems ACE XML Gateway and Manager Common Criteria Specific Functional Test Specification*
- *Cisco Systems ACE XML Gateway and Manager Test Coverage Reports*
- *Cisco Systems ACE XML Gateway and Manager Test Result Reports*

The Tests assurance measure satisfies the following EAL 3 augmented with ALC_FLR.2 assurance requirements:

- ATE_COV.1
- ATE_FUN.1
- ATE_IND.2
- ATE_DPT.1

6.2.7 Vulnerability assessment

Cisco Systems has conducted a strength of function analysis wherein all permutational or probabilistic security mechanisms have been identified and analyzed resulting in a demonstration that all of the relevant mechanisms fulfill the minimum strength of function claim, SOF-basic.

Cisco Systems performs regular vulnerability analyses of the entire TOE to identify weaknesses that can be exploited in the TOE.

These activities are documented in:

- *Cisco Systems ACE XML Gateway and Manager Version 5.0.3 Vulnerability Assessment Procedures*

The Vulnerability assessment assurance measure satisfies the following EAL 3 augmented with ALC_FLR.2 assurance requirements:

- AVA_SOF.1
- AVA_VLA.1

7 Protection Profile Claims

There is no Protection Profile claim in this Security Target.

8 Rationale

This section provides the rationale for completeness and consistency of the Security Target. The rationale addresses the following areas:

- Security Objectives;
- Security Functional Requirements;
- Security Assurance Requirements;
- Strength of Functions;
- Requirement Dependencies;
- TOE Summary Specification; and,
- PP Claims.

8.1 Security Objectives Rationale

This section shows that all secure usage assumptions, organizational security policies, and threats are completely covered by security objectives. In addition, each objective counters or addresses at least one assumption, organizational security policy, or threat.

8.1.1 Complete Coverage – Threat and Organizational Security Policies

This section demonstrates how there is at least one TOE or IT environment objective for each threat and/or organizational security policy, and how each objective can be traced back to at least one policy or threat. The following table shows this mapping, and the table is followed by a discussion of the coverage for each policy and threat.

	O.ACCOUN	**O.AUDREC**	**O.IDAUTH**	**O.SECFUN**	**O.SELPRO**	**O.LIMEXT**	**OE.SINGEN**	**OE.ENCRYP**	**O.TIME**
P.ACCOUN	X								X
P.AUDREC		X							
P.ENCRYP								X	
P.IDAUTH			X				X		
P.SECFUN				X					
P.LIMEXT						X			
T.SELPRO					X				

Table 4: Complete Coverage – Threat and Organizational Security Policies

8.1.1.1 P.ACCOUN

> *The TOE must provide user accountability for information flows through the TOE and for all use of security functions related to audit.*

This Policy is satisfied by ensuring that:

- O.ACCOUN: The TOE provides auditing (using time stamps from the operating system in the IT environment), requires users first identify themselves, allows only authorized administrators and users with appropriate security roles to access security functions related to audit, and generates an audit trail entry for each use of any TOE security function related to audit.
- O.TIME: Audited events are time-stamped.

8.1.1.2 P.AUDREC

The TOE must provide a readable audit trail of security-related events.

This Policy is satisfied by ensuring that:

- O.AUDREC: This is a restatement of policy. The TOE provides administrator console interfaces to read audit records.

8.1.1.3 P.ENCRYP

The TOE must protect the confidentiality of its dialogue with an authorized administrator through encryption, if the TOE allows administration to occur remotely from a connected network.

This Policy is satisfied by ensuring that:

- OE. ENCRYP: The hardware cryptomodule that is installed as a component of the hardware server appliance is used for digital signatures, hashing, SSL, encryption; specifically, the TOE uses SSL to encrypt the connection, keys, and TSF data that travels between an authorized administrator's Web browser and the Manager, and between the Manager and the Gateway.

8.1.1.4 P.IDAUTH

The TOE must uniquely identify and authenticate the claimed identity of all users, before granting a user access to TOE functions or, for certain specified services, to a connected network.

This Policy is satisfied by ensuring that:

- O.IDAUTH: This is a restatement of policy. The TOE supports individual users, which it requires to be identified and authenticated using supported authentication mechanisms.
- OE.SINGEN: All Web service traffic passes through the TOE.

8.1.1.5 P.SECFUN

The TOE must provide functionality that enables an authorized administrator or user with appropriate security roles to use the TOE security functions, and must ensure that only authorized administrators or users with appropriate security roles are able to access such functionality.

This Policy is satisfied by ensuring that:

- O.SECFUN: This is a restatement of policy. The TOE administrator console restricts access to the audit trail using its interfaces, the TOE supports individual users, the TOE provides Manager GUI interfaces to manage its security functions.

8.1.1.6 P.LIMEXT

The TOE must provide the means for an authorized administrator to control and limit access to TOE security functions by an authorized external IT entity.

This Policy is satisfied by ensuring that:

- O.LIMEXT: This is a restatement of policy. The TOE restricts the ability to specify Web services to administrators.

8.1.1.7 T.SELPRO

The TOE must protect itself against attempts by unauthorized users to bypass, deactivate, or tamper with TOE security functions.

This Threat is satisfied by ensuring that:

- O.SELPRO: This is a restatement of threat. The TOE restricts access to both Gateway and Manager component interfaces, disables accounts after failed attempts, and restricts access to the audit trail using its administrative interfaces. On startup, the TOE adopts a restrictive default state before attempting to return to normal operation. Errors achieving this state or loading the current Web Services SFP cause the TOE to enter a maintenance mode and log the errors. The TOE uses SSL to prevent tampering with TSF data in transit between separate parts of the TOE.

8.1.2 Complete Coverage – Environmental Assumptions

This section provides evidence demonstrating coverage of the Non-IT security objectives by the environmental assumptions. The following table shows this assumption to objective mapping.

		OE.INSTAL	OE.PHYCAL	OE.CREDEN	OE.PERSON	OE.INTROP	OE.SINGEN
Intended usage assumptions	**A.ACCESS**					X	
	A.ASCOPE					X	
	A.DYNMIC				X	X	
	A.SINGEN						X
Physical assumptions	**A.LOCATE**		X				
	A.PROTCT		X				
Personnel assumptions	**A.MANAGE**				X		
	A.NOEVIL	X	X	X			

Table 5: Complete coverage – environmental assumptions

8.1.2.1 A.ACCESS

The TOE has access to all the IT System data it needs to perform its functions.

This Assumption is satisfied by ensuring that:

- OE.INTROP: The OE.INTROP objective ensures the TOE has the needed access.

8.1.2.2 A.ASCOPE

The TOE is appropriately scalable to the IT System the TOE monitors.

This Assumption is satisfied by ensuring that:

- OE.INTROP: The OE.INTROP objective ensures the TOE has the necessary interactions with the IT System it monitors.

8.1.2.3 A.DYNMIC

The TOE will be managed in a manner that allows it to appropriately address changes in the IT System the TOE monitors.

This Assumption is satisfied by ensuring that:

- OE.INTROP: The OE.INTROP objective ensures the TOE has the proper access to the IT System.
- OE.PERSON: The OE.PERSON objective ensures that the TOE will managed appropriately.

8.1.2.4 A.SINGEN

Information can not flow among the internal and external networks unless it passes through the TOE.

This Assumption is satisfied by ensuring that:

- OE.SINGEN: The OE. SINGEN objective ensures the TOE mediates traffic between the internal and external networks.

8.1.2.5 A.LOCATE

The processing resources of the TOE will be located within controlled access facilities, which will prevent unauthorized physical access.

This Assumption is satisfied by ensuring that:

- OE.PHYCAL: The OE.PHYCAL provides for the physical protection of the TOE.

8.1.2.6 A.PROTCT

The TOE hardware and software critical to security policy enforcement will be protected from unauthorized physical modification.

This Assumption is satisfied by ensuring that:

- OE.PHYCAL: The OE.PHYCAL provides for the physical protection of the TOE.

8.1.2.7 A.MANAGE

There will be one or more competent individuals assigned to manage the TOE and the security of the information it contains.

This Assumption is satisfied by ensuring that:

- OE.PERSON: The OE.PERSON objective ensures all authorized administrators are qualified and trained to manage the TOE.

8.1.2.8 A.NOEVIL

The authorized administrators are not careless, willfully negligent, or hostile, and will follow and abide by the instructions provided by the TOE documentation.

This Assumption is satisfied by ensuring that:

- OE.INSTAL: The OE.INSTAL objective ensures that the TOE is properly installed and operated.
- OE.PHYCAL: The OE.PHYCAL objective provides for physical protection of the TOE by authorized administrators.
- OE.CREDEN: The OE.CREDEN objective supports this assumption by requiring protection of all authentication data.

8.2 Security Requirements Rationale

This section provides evidence supporting the internal consistency and completeness of the components (requirements) in the Security Target. Note that **Table 6: Objective to Requirement Correspondence** indicates the requirements that effectively satisfy the individual objectives.

8.2.1 Security Functional Requirements Rationale

All Security Functional Requirements (SFR) identified in this Security Target are fully addressed in this section and each SFR is mapped to the objective for which it is intended to satisfy.

	O.ACCOUN	**O.AUDREC**	**O.IDAUTH**	**O.SECFUN**	**O.SELPRO**	**O.LIMEXT**	**O.TIME**	**OE.ENCRYP**
FAU_GEN.1	X							
FAU_SAR.1.1a, b, c		X						
FAU_STG.1				X	X			
FCS_CKM.1								X
FCS_COP.1								X
FDP_IFC.1	X		X					
FDP_IFF.1	X		X					
FIA_AFL.1					X			
FIA_ATD.1			X	X				
FIA_UAU.2			X					
FIA_UID.2	X		X					
FMT_MOF.1				X	X	X		
FMT_MSA.1.1a,b,c	X			X				
FMT_MSA.3					X			
FMT_MTD.1.1a, b, c, d, e	X			X				
FMT_REV.1				X				
FMT_SMF.1				X				
FMT_SMR.1				X				
FPT_ITT.1						X		
FPT_RVM.1					X			
FPT_STM.1							X	

Table 6: Objective to Requirement Correspondence

8.2.1.1 O.ACCOUN

The TOE must provide user accountability for all attempts at authentication, for all attempted information flows through the TOE, for all attempted use of TSF-mediated security-management or audit functions, and for all modifications to the values of TSF data.

This TOE Security Objective is satisfied by ensuring that:

- FIA_UID.2: The TOE offers no TSF-mediated functions until the user is identified.
- FDP_IFC.1: The TOE offers no functions mediated by the Web services SFP unless all subject and information security attributes required for the requested information flow are satisfied.
- FDP_IFF.1: The TOE allows no information flows between subjects or objects unless the requested information flow unambiguously satisfies all of the requirements of the Web services SFP that an authorized administrator has defined for the requested operation.
- FMT_MSA.1: The TOE restricts the ability to modify or delete subject security attributes, information security attributes, and user security attributes to authorized administrators having sufficient roles. The TOE restricts the ability to view subject security attributes to users that have the Policy View role and authorized administrators. The TOE restricts the ability to view information security attributes to users that have the External Developer role, users that have the Policy View role, and authorized administrators.
- FAU_GEN.1: The TOE generates and records audit events for startup and shutdown of the audit function, all auditable events for the minimum level of audit, and for all attempted or successful modifications of TSF data.
  Note: The audit function starts automatically when the TOE starts up and the audit function shuts down automatically when the TOE shuts down. The TOE does not provide the capability to disable audit functionality. The audit function always runs when the TOE runs.
- FMT_MTD.1: The TOE restricts the ability to initialize and set user authentication data to authorized administrators having sufficient roles. The TOE restricts the ability to modify and reset an account’s own password to authorized administrators and users. The TOE restricts the ability to view or query audit records to authorized administrators having sufficient roles, with the exception that users who have the Policy View role can query the Event Log and users who have the Message Traffic Log role can query the Message Traffic Log.
- FPT_STM.1: The operating system provides a reliable timestamp.

8.2.1.2 O.AUDREC

The TOE must provide a readable audit trail of security-related events.

This TOE Security Objective is satisfied by ensuring that:

- FAU_SAR.1: The TOE provides Manager GUI interfaces that can be used to read audit trail files.

8.2.1.3 O.IDAUTH

The TOE must uniquely identify and authenticate the claimed identity of all users, before granting a user access to TOE functions or, for certain specified services, to a connected network.

This TOE Security Objective is satisfied by ensuring that:

- FIA_ATD.1: The TOE maintains user identities, authentication data for supported authentication mechanisms, and role information.
- FIA_UAU.2: The TOE offers no TSF-mediated functions until the user is authenticated.
- FDP_IFC.1: The TOE requires subjects to authenticate per FDP_IFF.1 before initiating an information flow subject to all rules the Web Services SFP defines.
- FDP_IFF.1: The TOE supports multiple types of subject and information security attributes that the WEB SERVICES SFP may use to authenticate subjects of the WEB SERVICES SFP according to the configuration specified for a given Web service.
- FIA_UID.2: The TOE offers no TSF-mediated functions until the user is identified.

8.2.1.4 O.SECFUN

The TOE must provide functionality that enables an authorized administrator or user with appropriate security roles to use the TOE security functions, and must ensure that only authorized administrators or users with appropriate security roles are able to access such functionality.

This TOE Security Objective is satisfied by ensuring that:

- FMT_MSA.1: The TOE restricts the ability to modify or delete subject security attributes, information security attributes, and user security attributes to authorized administrators having sufficient roles. The TOE restricts the ability to view subject security attributes to users that have the Policy View role and authorized administrators. The TOE restricts the ability to view information security attributes to users that have the External Developer role, users that have the Policy View role, and authorized administrators.
- FAU_STG.1: The TOE permits only authorized administrators and users who have been assigned appropriate security roles to access the TOE directly after successful identification and authentication to the Manager GUI. The Manager GUI does not provide any way to edit or delete the files that compose the audit trail.
- FIA_ATD.1: The TOE maintains user identities, authentication data for supported authentication mechanisms, and role information.
- FMT_MOF.1: The TOE restricts the ability to assign roles required to specify Web services policy to the ConsoleAdmin authorized administrator.
- FMT_MTD.1: The TOE restricts the ability to initialize and set user authentication data to authorized administrators having sufficient roles. The TOE restricts the ability to modify and reset an account’s own password to authorized administrators and users. The TOE restricts the ability to view or query audit records to authorized administrators having sufficient roles, with the exception that users who have the Policy View role can query the Event Log and users who have the Message Traffic Log role can query the Message Traffic Log.
- FMT_REV.1: The TOE restricts the ability to revoke user security attributes to authorized administrators by restricting access to user security attribute editors in the Manager to authorized administrators.
- FMT_SMF.1: The TOE provides authorized administrators with the ability to manage Web services, to manage users, and to manage the audit trail using the Manager.
- FMT_SMR.1: The TOE defines a login account of type Privileged User that can be assigned zero or more system-defined roles. Any Privileged User account that is assigned one or more of the system-defined roles “Operations”, “Access Control”, or “Routing” is considered an “authorized administrator.” The single factory-configured `administrator` account always has all roles, cannot be modified or deleted, and is considered an “authorized administrator.”

8.2.1.5 O.SELPRO

TOE must protect itself against attempts by unauthorized users to bypass, deactivate, or tamper with TOE security functions.

This TOE Security Objective is satisfied by ensuring that:

- FAU_STG.1: The TOE requires authentication using its local account authentication mechanism before making any TSFs available, including those related to the audit trail; therefore, unauthorized users cannot bypass, deactivate, or tamper with any TSFs, including those related to auditing functions.
- FIA_AFL.1: The TOE disables accounts after three failed login attempts to the Manager.
- FPT_RVM.1: The WEB SERVICES SPF cannot be bypassed by consumers. Similarly, both Gateway and Manager interfaces are restricted to authorized administrators and users.
- FMT_MOF.1: The TOE restricts the ability to assign roles required to specify Web services policy to the ConsoleAdmin authorized administrator.

- FMT_MSA.3: The TOE provides restrictive default values for security attributes used to enforce the WEB SERVICE SFP. The TOE also allows authorized administrators to specify alternative initial values.

8.2.1.6 O.LIMEXT

The TOE must provide the means for an authorized administrator to control and limit access to TOE security functions by an authorized external IT entity.

This TOE Security Objective is satisfied by ensuring that:

- FMT_MOF.1: The TOE restricts the ability to assign roles required to specify Web services policy to the ConsoleAdmin authorized administrator.
- FPT_ITT.1: The TOE uses SSL when managing the Gateway using the Manager to protect TSF data from disclosure.

8.2.1.7 O.TIME

The TOE will provide a time source that provides reliable time stamps.

This TOE Security Objective is satisfied by ensuring that:

- FPT_STM.1: The operating system provides a reliable timestamp.

8.2.1.8 OE.ENCRYP

The IT environment will protect the confidentiality of its dialogue with an authorized administrator through encryption when performing remote administration from a connected network.

This TOE Security Objective is satisfied by ensuring that:

- FCS_CKM.1: The cryptomodule in the IT environment can generate keys for use in digital signature, encryption, and SSL operations.
- FCS_COP.1: The TOE cryptomodule in the IT environment can perform hashing, digital signature, encryption, and SSL operations.

8.3 Security Assurance Requirements Rationale

The TOE is targeted at a generalized IT environment with good physical access security and competent administrators. Within such environments it is assumed that attackers will have a low to moderate attack potential. As such, EAL 3 (augmented with ALC_FLR.2) is appropriate to provide the assurance necessary to counter the potential for attack.

8.4 Strength of Functions Rationale

The overall strength of function claim of basic is believed to be commensurate with the overall assurance claim of EAL 3 augmented with ALC_FLR.2. The only applicable security function is Identification and Authentication where passwords are used by users as evidence of their claimed identities. The intent is that the password mechanism meets or exceeds basic and the evidence can be found in the strength of function analysis included in the *Cisco Systems ACE XML Gateway and Manager Version 5.0.3 Vulnerability Assessment Procedures* document.

8.5 Requirement Dependency Rationale

The following table demonstrates that all dependencies among the claimed security requirements are satisfied and therefore the requirements work together to accomplish the overall objectives defined for the TOE.

ST Requirement	CC Dependencies	ST Dependencies
FAU_GEN.1	FPT_STM.1	FPT_STM.1
FAU_SAR.1.1a, b, c	FAU_GEN.1	FAU_GEN.1
FAU_STG.1	FAU_GEN.1	FAU_GEN.1
FCS_CKM.1	(FCS_CKM.2 or FCS_COP.1) and FCS_CKM.4 and FMT_MSA.2	FCS_COP.1 and FCS_CKM.4. See note 1 below.
FCS_CKM.4	(FDP_ITC.1 or FCS_CKM.1) and FMT_MSA.2	FCS_CKM.1 See note 1 below.
FCS_COP.1	(FDP_ITC.1 or FCS_CKM.1) and FCS_CKM.4 and FMT_MSA.2	FCS_CKM.1 and FCS_CKM.4 See note 1 below.
FDP_IFC.1	FDP_IFF.1	FDP_IFF.1
FDP_IFF.1	FDP_IFC.1 and FMT_MSA.3	FDP_IFC.1 and FMT_MSA.3
FIA_AFL.1	FIA_UAU.1	FIA_UAU.2
FIA_ATD.1	none	none
FIA_UAU.2	FIA_UID.1	FIA_UID.2
FIA_UID.2	none	none
FMT_MSA.1.1a, b, c	(FDP_ACC.1 or FDP_IFC.1) and FMT_SMF.1 and FMT_SMR.1	FDP_IFC.1 FMT_SMF.1 and FMT_SMR.1
FMT_MSA.3	FMT_MSA.1 and FMT_SMR.1	FMT_MSA.1 and FMT_SMR.1
FMT_MOF.1	FMT_SMR.1 and FMT_SMF.1	FMT_SMR.1 and FMT_SMF.1
FMT_MTD.1.1a, b, c, d, e	FMT_SMR.1 and FMT_SMF.1	FMT_SMR.1 and FMT_SMF.1
FMT_REV.1	FMT_SMR.1	FMT_SMR.1
FMT_SMR.1	FIA_UID.1	FIA_UID.2
FPT_ITT.1	none	none
FPT_RVM.1	none	none
FPT_STM.1	none	none
ACM_CAP.3	ALC_DVS.1	ALC_DVS.1
ACM_SCP.1	none	none
ADO_DEL.1	none	none
ADO_IGS.1	AGD_ADM.1	AGD_ADM.1
ADV_FSP.1	ADV_RCR.1	ADV_RCR.1
ADV_HLD.2	ADV_FSP.1 and ADV_RCR.1	ADV_FSP.1 and ADV_RCR.1
ADV_RCR.1	none	none
AGD_ADM.1	ADV_FSP.1	ADV_FSP.1
AGD_USR.1	ADV_FSP.1	ADV_FSP.1
ALC_DVS.1	none	none
ALC_FLR.2	none	none
ATE_COV.2	ADV_FSP.1 and ATE_FUN.1	ADV_FSP.1 and ATE_FUN.1
ATE_FUN.1	ADV_FSP.1 and AGD_ADM.1 and AGD_USR.1	ADV_FSP.1 and AGD_ADM.1 and AGD_USR.1
ATE_DPT.1	ADV_HLD.1 and ATE_FUN.1	ADV_HLD.1 and ATE_FUN.1
ATE_IND.2	ADV_FSP.1 and AGD_ADM.1 and AGD_USR.1 and ATE_FUN.1	ADV_FSP.1 and AGD_ADM.1 and AGD_USR.1 and ATE_FUN.1
AVA_SOF.1	ADV_FSP.1 and ADV_HLD.1	ADV_FSP.1 and ADV_HLD.1
AVA_VLA.1	ADV_FSP.1 and ADV_HLD.1 and	ADV_FSP.1 and ADV_HLD.1 and

	AGD_ADM.1 and AGD_USR.1	AGD_ADM.1 and AGD_USR.1
AVA_MSU.1	ADO_IGS.1 and ADV_FSP.1 and AGD_ADM.1 and AGD_USR.1	ADO_IGS.1 and ADV_FSP.1 and AGD_ADM.1 and AGD_USR.1

Note 1: Functional component FCS_COP.1 depends on the following functional components: FCS_CKM.1 Cryptographic key generation, FCS_CKM.4 Cryptographic key destruction, and FMT_MSA.2 Secure Security Attributes. The cryptographic module in the IT environment is FIPS PUB 140-2 validated. The TOE initializes this module in "strict FIPS mode" and uses this module exactly as specified by the validation testing; therefore, the dependencies of key generation, key destruction, and secure key values are satisfied by this module's validation as FIPS PUB 140-2 compliant. For more information, refer to FIPS PUB 140-2, sections 4.7.2 Key Generation and 4.7.6 Key Zeroization.

8.6 Explicitly Stated Requirements Rationale

There are no explicitly stated requirements in this Security Target.

8.7 TOE Summary Specification Rationale

This Section, in conjunction with Section 6, the TOE Summary Specification, provides evidence that the TOE security functions are suitable to meet the TOE security requirements. Each subsection in Section 6, the TOE Summary Specification, describes a security function of the TOE. Each description is followed with rationale that indicates which requirements are satisfied by aspects of the corresponding security function. The set of security functions work together to satisfy all of the security functions and assurance requirements. Furthermore, all of the security functions are necessary in order for the TSF to provide the required security functionality.

Table 7 Security Functions vs. Requirements Mapping demonstrates the relationship between security requirements and security functions.

	Security Audit	**User data protection**	**I&A**	**Security management**	**TSF Protection**
FAU_GEN.1	X				
FAU_SAR.1.1a, b, c	X				
FAU_STG.1	X				
FDP_IFC.1		X			
FDP_IFF.1		X			
FIA_AFL.1			X		
FIA_ATD.1			X		
FIA_UAU.2			X		
FIA_UID.2			X		
FMT_MOF.1				X	
FMT_MSA.1.1a, b, c				X	
FMT_MSA.3				X	
FMT_MTD.1.1a, b, c, d, e				X	
FMT_REV.1				X	
FMT_SMF.1				X	
FMT_SMR.1				X	
FPT_ITT.1					X
FPT_RVM.1					X
FPT_STM.1					X

Table 7 Security Functions vs. Requirements Mapping

8.8 PP Claims Rationale

See Section 7, Protection Profile Claims.