LogRhythm Integrated Solution

Security Target

Version 1.1
March 30, 2012

Prepared for:
LogRhythm, Inc.
4780 Pearl East Circle
Boulder, CO 80301

Prepared By:
Science Applications International Corporation
Common Criteria Testing Laboratory
6841 Benjamin Franklin Drive.
Columbia, MD 21046

Warranty

The information contained in this document is subject to change without notice. LogRhythm, Inc. makes no warranty of any kind with respect to this information. LogRhythm, Inc. specifically disclaims the implied warranty of the merchantability and fitness for a particular purpose. LogRhythm, Inc. shall not be liable for any direct, indirect, incidental, consequential, or other damage alleged in connection with the furnishing or use of this information.

Trademark

LogRhythm® is a trademark of LogRhythm, Inc.

Contents

Tables

1. Security Target Introduction

This section identifies the Security Target (ST) and Target of Evaluation (TOE) identification, ST conventions, ST conformance claims, and the ST organization.

The TOE is an Intrusion Detection System (IDS) consisting of several components that coordinate with one another to collect and analyze information from multiple log sources (such as Windows events, syslog, flat file, NetFlow, sFlow, databases or applications) and provides tools to view and analyze IDS results and to issue alerts of significant events.

The product can also provide endpoint monitoring and control functionality, but these capabilities are not addressed by the IDS System PP, to which the TOE claims conformance. As such, they are outside the scope of the evaluation. The endpoint monitoring and control functionality is provided by three components: the User Activity Monitor (UAM); the File Integrity Monitor (FIM); and the Data Loss defender (DLD). All three components are disabled by default. In addition, the FIM requires a separate license.

The product includes support for the use of a SQL Server for user authentication, but this option must be disabled in the evaluated configuration. The TOE must be configured to use Windows Active Directory or the local Windows operating system for user authentication.

The Security Target contains the following additional sections:

- TOE Description (Section 2)

  This section gives an overview of the TOE, describes the TOE in terms of its physical and logical boundaries, and states the scope of the TOE.

- Security Problem Definition (Section 3)

  This section details the expectations of the environment, including the assumptions, organizational security policies, and threats that are countered by the TOE and TOE environment.

- Security Objectives (Section 4)

  This section details the security objectives of the TOE and TOE environment.

- IT Security Requirements (Section 5)

  This section presents the security functional requirements (SFRs) for the TOE, and details the assurance requirements for EAL2 augmented with ALC_FLR.2.

- TOE Summary Specification (Section 6)

  This section describes the security functions represented in the TOE that satisfy the security requirements.

- Protection Profile Claims (Section 7)

  This section presents the Protection Profile claims and supporting rationale.

- Rationale (Section 8)

  This section closes the ST with the justifications of the security objectives, requirements and TOE summary specifications as to their consistency, completeness, and suitability.

1.1 Security Target, TOE and CC Identification

ST Title – LogRhythm Integrated Solution Security Target

ST Version – Version 1.1

ST Date –March 30, 2012

TOE Identification – LogRhythm, Version 6.0.4 with Microsoft SQL Server 2008 R2 Enterprise Edition

TOE Developer – LogRhythm, Inc.

Evaluation Sponsor – LogRhythm, Inc.

CC Identification – Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 2, September 2007

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 3.1, Revision 2, September 2007.
    - Part 2 Extended
- Common Criteria for Information Technology Security Evaluation Part 3: Security Assurance Components, Version 3.1, Revision 2, September 2007.
    - Part 3 Conformant
- This ST and the TOE it describes are conformant to the following package:
    - EAL 2 Augmented with ALC_FLR.2
- This ST and the TOE it describes are conformant to the following protection profile:
    - *U.S. Government Protection Profile Intrusion Detection System System for Basic Robustness Environments,* Version 1.7, July 25, 2007 augmented with FIA_UID.2, FIA_UAU.2 and FMT_SMF.1.

1.3 Conventions and Acronyms

This section specifies the formatting information used in the Security Target.

1.3.1 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 placed at the end of the component. For example FDP_ACC.1(1) and FDP_ACC.1(2) indicate that the ST includes two iterations of the FDP_ACC.1 requirement, (1) and (2).
    - Assignment: allows the specification of an identified parameter. Assignments are indicated using bold and are surrounded by brackets (e.g., [**assignment**]). Note that an assignment within a selection would be identified in italics and with embedded bold brackets (e.g., [***[selected-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 …").
    - Extended Requirements are allowed to create requirements should the Common Criteria not offer suitable requirements to meet the ST needs. To ensure these requirements are explicitly identified, the ending "_(EXT)" is appended to the newly created short name and the component.

- Other sections of the ST – Other sections of the ST use bolding to highlight text of special interest, such as captions.

1.3.2 Acronyms

AI	Advanced Intelligence
CC	Common Criteria
CEM	Common Evaluation Methodology
CCEVS	Common Criteria Evaluation and Validation Scheme
CSP	Cryptographic Service Provider
DoD	Department of Defense
EAL	Evaluation Assurance Level
EM	Event Manager
GUI	Graphical User Interface
HLD	High-level Design
IA	Initial Assessment
IDS	Intrusion Detection System
LM	Log Manager
LOI	Letter of Interest
NIAP	National Information Assurance Partnership
NIST	National Institute of Standards and Technology
NSA	National Security Agency
PP	Protection Profile
R2	Release 2
SAIC	Science Applications International Corporation
SAR	Security Assurance Requirement
SLF	Agent-only Log Collection Appliance
SFR	Security Functional Requirement
SP	Service Pack
SSL	Secure Sockets Layer
ST	Security Target
TDS	Tabular Data Stream
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Function(s)
TSFI	TOE Security Function Interface(s)
US	United States

2. TOE Description

The Target of Evaluation (TOE) is LogRhythm 6.0.4 with Microsoft SQL Server 2008 R2 Enterprise Edition. The TOE collects, categorizes, identifies, and normalizes log data from log sources such as Windows events, syslog, flat file, NetFlow, sFlow, databases, and applications, and provides automated alerting capabilities. The TOE can detect security and compliance issues, such as anomalies in authentication activity, and brute force attacks on monitored servers.

The TOE provides automated centralization of log collection, archival and recovery, automated reporting, forensic investigation abilities, anomaly and insider threat detection, turnkey appliance configuration, and a console management interface.

2.1 TOE Overview

A deployment of LogRhythm consists of:

- 1 Event Manager
- 0 or more Advanced Intelligence Engine (AI Engine) Servers(s)
- 1 or more Log Manager(s)
- 1 or more System Monitor Agent(s) with Trace File Converter
- 1 or more Console(s)
- 1 or more SQL Server instances.

The Event Manager (EM) and Log Manager (LM) can reside on the same server for low-volume deployments, or on dedicated servers for high volume deployments. Each AI Engine Server is always a standalone server. A SQL Server instance resides on each Log Manager server and on the Event Manager server. The System Monitor Agents can be deployed on Windows, Linux, Solaris, HP-UX or AIX systems. The Console is a Windows application that runs on an administrator's workstation.

2.2 TOE Architecture

The following figure depicts the TOE components within their environment and shows communications among the components and operational environment devices. SQL Server is an internal component of Log Managers and Event Manager, and is not shown in the figure.

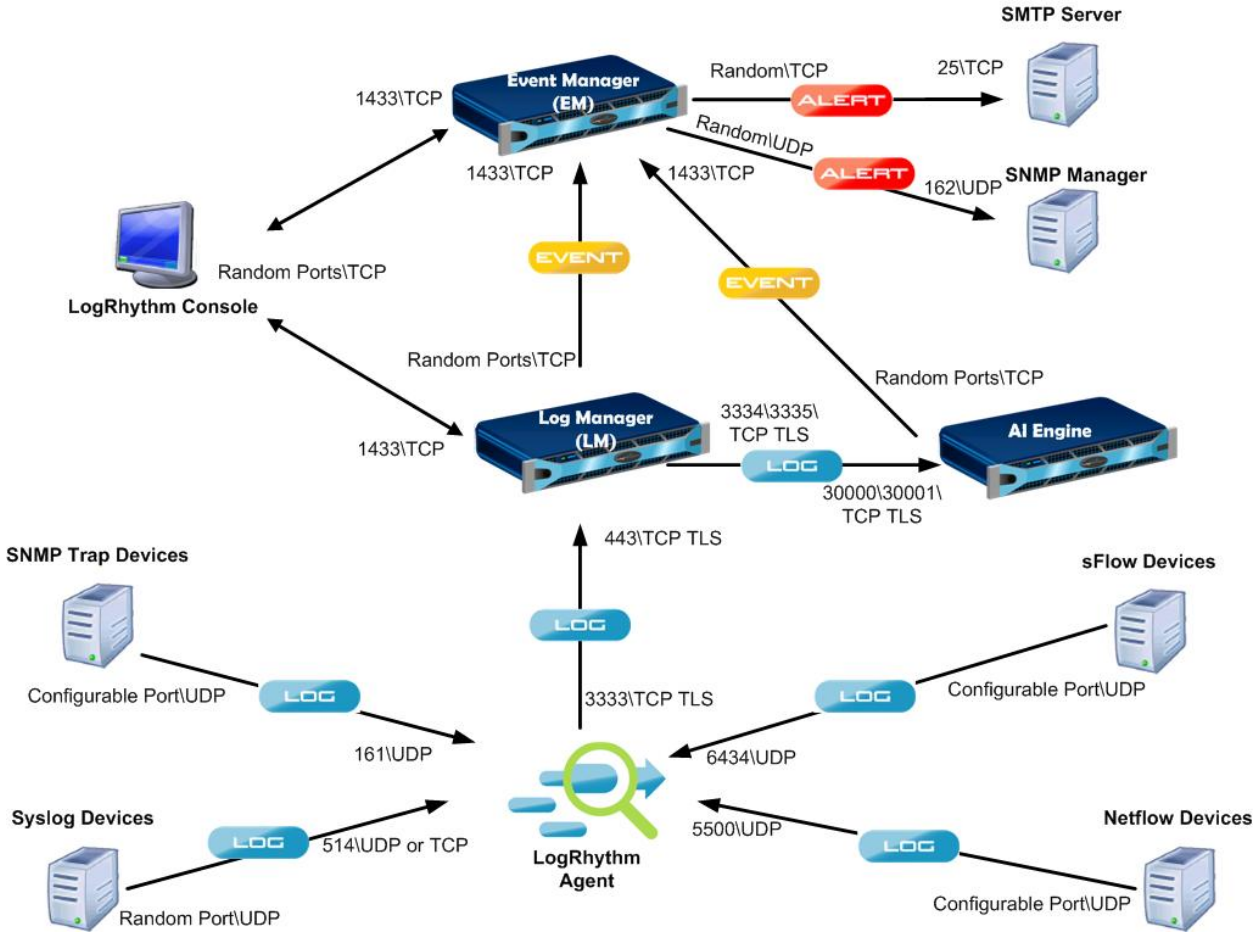

Figure 1 TOE Components in Operational Environment

The LogRhythm System Monitor Agent(s)[1], Log Manager(s), AI Engine Server(s), Event Manager, Console(s) and SQL Server software constitute the TOE. The TOE can be purchased as software only or pre-configured on dedicated appliances. These options are described in detail in Section 2.2.1 below.

In general, log information flows from System Monitor Agents through Log Managers and AI Engine Servers to the Event Manager with SQL Server used internally to store log information. System Monitor Agents collect log messages. Log Managers analyze individual log messages and identify Events. An Event is a log message or collection of log messages that LogRhythm determines to be important or interesting. AI Engine Servers analyze log metadata gleaned from sets of log messages to identify more complex Events. The Event Manager processes Events and raises alarms as appropriate. Administrators use the Console to configure LogRhythm (for example, selecting rules that identify Events) and to view log reports and analyses.

The System Monitor Agents are capable of collecting logs from most sources including Windows events (local and remote), syslog, flat file, NetFlow, databases, or applications. The System Monitor Agent converts collected logs to ASCII text strings, which can be encrypted before forwarding across untrusted networks (e.g. Internet). Trace File Converter is a support service, which converts binary SQL Server log data to UTF-8 text suitable for collection by a System Monitor Agent

Each System Monitor Agent forwards logs to the LM that is configured to receive them, where they are analyzed against defined Knowledge Base rules, written to a centralized database in the LM, and also archived on a file system. System Monitor Agent communications with LM(s) are authenticated and encrypted via FIPS 140-2

[1] Labeled "LogRhythm Agent" in Figure 1.

certified SSL[2]. Each LM consists of a SQL Server 2008 R2 instance and a LogRhythm Mediator Server. The Mediator Server takes in log messages (collected and forwarded by LogRhythm System Monitor Agents) and processes them against Knowledge Base rules that identify and categorize the log messages. The applied Knowledge Base rules determine whether the Mediator Server forwards log metadata to an AI Engine or forwards the log message to the EM as an Event or both. The Mediator Server is also responsible for writing incoming logs to an active archive, which is a file on the file system of the LM Host. Once that active archive file reaches a certain size or age (administrative configurable), the active archive is converted to an inactive archive file. During that conversion, the contents are SHA-1 hashed and then compressed. The SHA-1 hash value is stored in a database table within the LogRhythm Event Manager. If there is a restore request of the logs contained within the inactive archives, the SHA-1 hash is verified to ensure that the file has not been altered since being sealed. Communications between LM and AI Engine Server and between LM and EM are protected by FIPS 140-2 certified SSL. Updates to the Knowledge Base rules can be obtained by licensed customers at the vendor's website.

An AI Engine Server consists of two services: AI Engine Communication Manager service and AI Engine service. The AI Engine Communication Manager receives log metadata from one or more Log Managers. It marshals the data for the AI Engine to process. Also, it maintains SSL connections with Log Managers. An AI Engine processes the data by applying AI rules to the set of log metadata collected over time. An AI rule can correlate multiple log messages to identify an Event, which the AI Engine sends to the EM.

The EM consists of two services: the LogRhythm Alarming and Response Manager (ARM) service and the Job Manager service together with a SQL Server instance. There is only one EM per deployment. The EM receives and maintains log information from the LMs that have been analyzed against the Knowledge Base rules and have been identified as Events. The EM receives Events corresponding to complex conditions from the AI Engine Server. The ARM service evaluates Alarm Rules to determine if an Event (or series of Events) should be alarmed on and, if so, what the response should be (e.g., sending e-mails to people on a notification list, sending SNMP traps, or perform a remediation action).

The Console provides the user interface into a LogRhythm deployment. The Console is a Windows .NET-based client application. Authenticated users can view logs, Events, alarms and reports. The Console also provides real-time monitoring, incident management, and interfaces for TOE configuration and user management. The Console tools also provide interfaces for the administrator to configure the pruning and aging scripts which are designed to automatically manage the LM and EM databases. All communications between the Console and LogRhythm servers (EM and LMs) are protected by FIPS 140-2 certified SSL. An AI Engine Server obtains its configuration from the Console indirectly via the EM.

Every TOE deployment will have one EM component, at least one LM component, and at least one System Monitor Agent component. An AI Engine Server is optional in a TOE deployment. Each appliance configuration includes agent software and, in the case where a software-only purchase is made, the agent software must also be installed. The System Monitor Agent is the only component responsible for collecting the logs. The System Monitor Agent collects the logs and forwards them to an LM. LogRhythm has the capability to perform "agent-based" or "agent-less" monitoring, dependent on the source of log data. Typically, monitoring flat files requires a System Monitor Agent be installed on the system where the log resides. This is an example of agent-based monitoring (i.e., any deployment where a System Monitor Agent must be installed where the log data resides). "Agent-less monitoring" refers to any log source that a System Monitor Agent accesses remotely via network resources (for example Windows Event logs via API or ASCII flat files via network storage shares) or any log source that pushes log messages to a System Monitor Agent (for example syslog and Checkpoint firewall logs). "Agent-less" does not mean there is no System Monitor Agent involved but rather the System Monitor Agent is not on the same system as the log source. Note that Windows Event Logs can be collected in an agent-based or agent-less configuration. In addition to being able to read its local Windows Event Logs, the Windows System Monitor Agent can connect to remote Windows assets and pull the Windows Event Logs, which are then forwarded to a LM. All System Monitor Agents contain an integrated syslog server. In addition, Windows System Monitor Agents include a SNMP trap receiver, Netflow servers, and sFlow Collectors, allowing for the reception of SNMP, Netflow, and sFlow data for collection.

[2] SSL is used as a generic term here. The TOE implements TLS 1.0. See section 6.1.4 Protection of the TSF.

2.2.1 Physical Boundaries

The TOE consists of the following software components:

- 1 Event Manager
- 0 or more AI Engine Server(s)
- 1 or more Log Manager(s)
- 1 or more System Monitor Agent(s)
- 1 or more Console(s)
- 1 or more SQL Server instances.

The Event Manager and Log Manager can reside on the same server for low-volume deployments, or on dedicated servers for high volume deployments. Each AI Engine Server runs on a dedicated system. The System Monitor Agents can be deployed on Windows, Linux, Solaris, HP-UX or AIX systems.

The Event Manager, AI Engine Server, Log Manager, and System Monitor Agent software can be pre-installed on vendor-supplied appliance(s) or can be installed directly on a system by the customer. Each appliance solution consists of LogRhythm software (EM, AI Engine Server, LM, System Monitor Agent, or Console), hardened Windows Operating System (Windows Server 2008 R2 with .NET framework) and hardened SQL Server 2008 R2 Enterprise Edition. Windows Server, .NET Framework, and the appliance hardware (if purchased in this configuration) are not part of the TOE. Each System Monitor Agent includes a syslog server. Additional syslog servers may be required to support additional log sources in the operational environment. SMTP servers are required to support the TOE.

The Console software can be installed on the following operating systems:

- Windows Server 2003 32- and 64-bit;
- Windows XP 32-bit;
- Windows Vista 32-bit– and 64-bit;
- Windows Server 2008 32– and 64-bit;
- Windows 7 32- and 64-bit; and
- Windows Server 2008 R2 64-bit.

All releases of the operating systems listed are supported. One or more of these operating systems is required for all configurations; one for each Console desired. Also required for notifications in all configurations is at least one SMTP or one SNMP server. A full-featured LogRhythm deployment would include servers for both types of notifications.

The integrated solution can be delivered in a single appliance called the XM appliance or through a combination of integrated Log Manager, AI Engine Server, and Event Manager appliances (called LM, AI Engine Server, and EM appliances, respectively). An XM appliance includes Log Manager and Event Manager, but not AI Engine Server. A deployment can contain XM, LM, AI Engine Server, and EM appliances and SLF (System Monitor Agent-only log collection) although only one Event Manager is active in a deployment.

An SLF (a stand-alone appliance that includes System Monitor Agent software) is not required for a typical deployment, as agents are generally installed directly on the customer's IT infrastructure. SLF's are generally indicated when change control prohibits installation of foreign software on customer systems.

LogRhythm appliances are available in two families of appliances. There are five different hardware configurations for LM, EM, and XM appliances: LRX1, LRX1-2, LRX2, LRX3, and LRX3-2. There are two different hardware configurations for AI Engine Server appliances: AIE1 and AIE2. Within each family, there are no security functional differences between the hardware configurations. The only differences are in performance and storage capacity.

For each LRX hardware configuration, three software configurations are available: EM appliance (Event Manager on one box), LM appliance (Log Manager on one box) or XM appliance (EM and LM on one box). In all three cases a SQL Server and System Monitor Agent are included on the appliance. The appliances with a designated -2 indicate the appliance provides twice the storage of the original model; i.e. LRX1-2 provides twice the storage of the LRX1. The LRX1 provides 12 GB RAM and 272 GB storage capacity. The LRX2 provides twice the RAM of the LRX1 and 834 GB storage capacity. The LRX3 provides 32GB RAM and 1.25 TB storage capacity.

Each AI Engine Server hardware configuration supports the same AI Engine Server software, as described above. The AIE1 provides 32 GB RAM and 272 GB storage capacity. The AIE2 provides 96 GB RAM and 544 GB storage capacity.

Optional System Monitor Agent-only collection appliances (SLFs), Log Manager appliances (LMs), and AI Engine Server appliances can be added as needed. The appliances can be configured to address log volumes ranging from tens of millions to over a billion logs per day. Customers expecting more logs may require additional LM and AI Engine Server appliances.

The TOE contains no dependencies on the underlying hardware and the appliance is provided to customers at their request only as a convenient packaging bundle. Regardless of whether the customer purchases a software-only solution or an appliance, the TOE executable is the same with the exception of the agent code which may differ based on the supported platform.

If a software only configuration is selected, the following operational environment components are required (in addition to the OS requirement for the Console identified above): one or more Windows Server 2008 R2 operating systems and the underlying hardware to host the EM, AI Engine Server, LM, and System Monitor Agent components. Note that the System Monitor Agents can additionally be installed on:

- Windows: XP 32-bit; Vista 32- and 64-bit; 7 32- and 64-bit; Server 2003 32- and 64-bit; Server 2008 32- and 64-bit; Server 2008 R2 64-bit;
- Linux 2.4: Red Hat Enterprise Linux (RHEL) 9 32-bit;
- Linux 2.6: CentOS 5.1 32-bit; CentOS 5.5 64-bit; Debian 5.0.3 32-bit; Fedora 7 32-bit; RHEL 5 32- and 64-bit; RHEL 6 64-bit; SUSE Linux 9 64-bit; Ubuntu 9.10 32- and 64-bit; Ubuntu 10 32-bit;
- Solaris 8, 9, and 10 SPARC; Solaris 10 x86;
- HP-UX 11i: v1, v2, v3 PA-RISC; v2, v3 Itanium 64-bit;
- AIX: 5.2, 5.3 and 6.1 64-bit.

All versions of the specified operating systems are supported. Depending on the supporting platform the System Monitor Agents will have different binaries, though the binaries for Linux, Solaris, HP-UX, and AIX share the same source code base. Also the TOE requires the following in the operational environment:

- Active Directory and
- .NET V3.5 (to support connections for Windows Server 2008 R2)

SMTP is provided by the OS and meets RFC821 specifications. Please see the LogRhythm Installation Guide for additional details including instructions for securing (hardening) the Windows operating system.

Moreover, the TOE may be run in a virtualized environment, since the TOE has no dependencies on the underlying hardware. The security environment is the same whether the customer installs TOE software on physical or virtual devices. Please see the LogRhythm Installation Guide for a Virtual Appliance installation of the TOE software.

Regardless of whether the customer purchases a software-only solution or an appliance, the TOE functionality is the same.

2.2.2 Logical Boundaries

This section identifies the security functions that LogRhythm provides. These comprise the following:

- Security Audit
- Identification and Authentication
- Security Management

- Protection of the TOE Security Functions
- IDS Component requirements (Both Analyzer and System).

The Console provides the user interface into a LogRhythm deployment. The Console is a Windows .NET-based client application. Authenticated users with the appropriate role can view logs, events, alarms and reports. The Console provides interfaces for TOE configuration and user management. Identification, Authentication and administrative activity are audited to hold users accountable for their actions. The audit records are protected from unauthorized modification and deletion. All administrators must be identified and authenticated (authentication is performed in the operational environment either by the local Windows OS or by Active Directory). The TOE enforces protected communications between the Console and LogRhythm servers (EM and LMs) using FIPS 140-2 certified SSL. The System Monitor Agents are capable of collecting multiple log source types from target systems and capable of recording information about these logs. In addition the logs are protected from unauthorized deletion or modification. A Log Manager provides signature and integrity analysis functions. An AI Engine Server provides additional signature functions. Alarms can be configured and sent to the alarm database and/or users configured to receive alarms. The system data can be viewed by authorized administrators and is protected from unauthorized modification and deletion.

The TOE protects itself from tampering and bypass through several mechanisms implemented by the TOE and the operational environment. The underlying operating system separates processes into separate domains and prevents one process from accessing memory space of another process. The TOE uses FIPS 140-2 validated SSL to protect data transmitted between the TOE components from unauthorized disclosure and modification. All SSL functionality is provided by the operating system in the operational environment.

The sections below summarize the security functions provided by the TOE.

2.2.2.1 Security Audit

The TOE recognizes the following events and is capable of collecting them:

- Startup and shutdown of the TOE's auditing function;
- Successful and unsuccessful attempts to read the audit records;
- Access to the TOE, the log records collected by the TOE, and events identified by the TOE;
- All use of identification and authentication mechanisms;
- Modifications in the behavior of the TOE security functions;
- Modifications to the values of TSF data; and
- Modifications to a user's security management role.

The TOE records various information about each audit record collected such as: the date and time of the event; the type of event; the subject identity; the outcome of the event; and other information specific to the event type. All security audit events are generated from the LogRhythm console. Other TOE components generate only operational and error logs.

The TOE provides an interface to authorized users to read audit records from the audit trail and this interface is restricted to authorized roles. The TOE provides the ability to sort audit records on various fields in the audit data, and to include or exclude auditable events from the set of audited events based on "event type". The TOE prevents unauthorized modifications and deletions to the stored audit records by minimizing the available interfaces and restricting these interfaces to the authorized authenticated administrator. In addition, the TOE prevents the loss of audit data in the event the space available for storing audit records is exhausted.

The TOE is a software only implementation and therefore relies on the operational environment to provide a reliable timestamp. Additionally, the audit logs are stored in the file system and therefore rely on the operational environment for protection of the logs due to file permission enforcement.

2.2.2.2 Identification and Authentication

LogRhythm requires all users to be identified and authenticated before accessing any TOE functionality through the Console. Users and roles are defined in the TOE, operating at the application layer. When a user logs in to the TOE, Windows Active Directory or the local Windows operating system authenticates the claimed user identity. Windows Active Directory and the local Windows operating system support both password and Common Access Card (CAC) credentials for user authentication. The TOE enforces the result. If authentication is successful then the application table is checked for the user's rights. If the user is not in the table then access is denied.

2.2.2.3 Security Management

The console provides the capability to manage the auditing, analysis and reaction functions. The management functions are restricted to administrative roles.

The TOE comes with two pre-defined administrative roles: Global Admin and Global Analyst. The TOE supports a customer-defined Restricted Analyst role (that is, subset of the Global Analyst privileges). These roles, when assigned to users, provide varying levels of access to the TOE interfaces and functions.

2.2.2.4 Protection of the TSF

All communication channels between TOE components are protected by FIPS 140-2 certified SSL. The TOE supports both self-signed certificates and user-supplied certificates for establishing SSL-protected communication. This includes the following communication channels:

- Console to Server (Event Manager or Log Manager) communications,
- System Monitor Agent to Log Manager communications,
- Log Manager to AI Engine Server communications,
- Log Manager to Event Manager communications, and
- AI Engine Server to Event Manager communications.

The integrity of LogRhythm archives is protected by SHA-1 hashing and compression. Logs received by the Log Manager are stored in an archive, which is a file on the file system of the Log Manager that is subsequently hashed and compressed by the Mediator Service. This protection is provided to inactive archived files for use in verifying integrity during archive restoration and other operations. The collected logs are formatted as ASCII text strings and can be encrypted before forwarding across untrusted networks (e.g. Internet). Modification of the archived logs can be detected by rehashing and comparing the values. Note that the SHA-1 hash values are stored in the EM database. Timestamps are provided by the operational environment. The TOE normalizes time stamps to account for time zone differences.

2.2.2.5 IDS Component requirements

The System Monitor Agents are able to collect relevant information from multiple sources. The Log Manager performs analysis on the collected information by processing the data against known signatures. The Log Manager forwards log metadata to the AI Engine Server. The AI Engine Server can analyze sets of logs for more complex signatures. For example, together the Log Manager and AI Engine Server can detect security event/violations based on integrity checks and signature definitions. The Event Manager can take the appropriate action such as writing the event to a log file or sending an alert to an administrator.

The analyzer and system logs and events can be viewed from the Console. A potential loss of logs can be prevented by the TOE's layered architecture by providing administrative interfaces to configure database sizes and automatic purging scripts.

Each log or event collected by the System Monitor Agent contains the date and time of the event (or log), subject identity, and the outcome of the event. In addition some logs contain location, service, protocol information; source and destination addresses; and other information specific to the type of log collected.

The System Monitor Agent is capable of collecting the following events:

- Start-up and shutdown,

- Identification and authentication events,
- Data accesses,
- Service requests,
- Network traffic,
- Security configuration changes,
- Data introduction,
- Detected malicious code,
- Access control configuration,
- Service configuration,
- Authentication configuration,
- Accountability policy configuration, and
- Detected known vulnerabilities.

A syslog server is included in each System Monitor Agent however the operational environment may be required to provide additional syslog servers to support additional log sources. Refer to Section 6 for more details.

2.2.3 Excluded Product Functionality

As noted in section 1, User Activity Monitor, File Integrity Monitor, and Data Loss defender provide functionality not addressed by the IDS System PP. All three components are disabled by default.

LogRhythm guidance documentation describes the following product features:

- High Availability,
- LogRhythm Backup and Recovery Procedures,
- Performance Counters,
- Log Processing Report,
- Network Visualization,
- Save Investigation as a Report,
- Reporting Center, and
- Customizing Reports.

These features were not covered by the evaluation.

Efficacy of custom remediation plug-ins was not covered by the evaluation, although the Auto Remediation Plug-ins provided by LogRhythm are within the scope of the evaluation and were tested.

TOE guidance documentation describes how to configure third-party devices to generate logs and how to configure the TOE to collect the logs. The third-party devices were not within the scope of evaluation.

2.3 TOE Documentation

LogRhythm offers installation and configuration guidance for the LogRhythm Integrated Solution product, as well as guidance for subsequent use and administration of the applicable security features. This guidance is available in the “LogRhythm Help” document.

3. Security Problem Definition

The TOE security environment describes the security aspects of the intended environment in which the TOE is to be used and the manner in which it is expected to be employed. The statement of the TOE security environment defines the following:

- Organizational Policies that the TOE and the environment of the TOE fulfill
- Threats that the TOE and the environment of the TOE counter
- Assumptions made about the operational environment and the intended method of use for the TOE.

Furthermore, the TOE is intended to be used in environments where the relative assurance that its security functions are enforced is commensurate with EAL 2 augmented with ALC_FLR.2 as defined in the CC.

All security environment statements have been drawn from a validated PP (*U.S. Government Protection Profile Intrusion Detection System System for Basic Robustness Environments,* Version 1.7, July 25, 2007 Protection Profile). Please consult this protection profile for the description of the security environment. The policies, threats and assumptions from that PP have been copied here for convenience; however, the IDS PP contains the definitive statement of security environment.

3.1 Assumptions

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

3.1.1 Intended Usage Assumptions

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

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

3.1.2 Physical Assumptions

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

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

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

A.NOTRST The TOE can only be accessed by authorized users.

3.2 Threats

The following are threats identified for the TOE and the IT System the TOE monitors. The TOE itself has threats and the TOE is also responsible for addressing threats to the environment in which it resides. The assumed level of expertise of the attacker for all the threats is unsophisticated.

3.2.1 TOE Threats

T.COMINT	An unauthorized user may attempt to compromise the integrity of the data collected and produced by the TOE by bypassing a security mechanism.
T.COMDIS	An unauthorized user may attempt to disclose the data collected and produced by the TOE by bypassing a security mechanism.
T.LOSSOF	An unauthorized user may attempt to remove or destroy data collected and produced by the TOE.
T.NOHALT	An unauthorized user may attempt to compromise the continuity of the System's collection and analysis functions by halting execution of the TOE.
T.PRIVIL	An unauthorized user may gain access to the TOE and exploit system privileges to gain access to TOE security functions and data.
T.IMPCON	An unauthorized user may inappropriately change the configuration of the TOE causing potential intrusions to go undetected.
T.INFLUX	An unauthorized user may cause malfunction of the TOE by creating an influx of data that the TOE cannot handle.
T.FACCNT	Unauthorized attempts to access TOE data or security functions may go undetected.

3.2.2 IT System Threats

The following identifies threats to the IT System that may be indicative of vulnerabilities in or misuse of IT resources.

T.SCNCFG	Improper security configuration settings may exist in the IT System the TOE monitors.
T.SCNMLC	Users could execute malicious code on an IT System that the TOE monitors which causes modification of the IT System protected data or undermines the IT System security functions.
T.SCNVUL	Vulnerabilities may exist in the IT System the TOE monitors.
T.FALACT	The TOE may fail to react to identified or suspected vulnerabilities or inappropriate activity.
T.FALREC	The TOE may fail to recognize vulnerabilities or inappropriate activity based on IDS data received from each data source.
T.FALASC	The TOE may fail to identify vulnerabilities or inappropriate activity based on association of IDS data received from all data sources.
T.MISUSE	Unauthorized accesses and activity indicative of misuse may occur on an IT System the TOE monitors.
T.INADVE	Inadvertent activity and access may occur on an IT System the TOE monitors.
T.MISACT	Malicious activity, such as introductions of Trojan horses and viruses, may occur on an IT System the TOE monitors.

3.3 Organizational Security Policies

An organizational security policy is a set of rules, practices, and procedures imposed by an organization to address its security needs. This section identifies the organizational security policies applicable to the TOE.

P.DETECT	Static configuration information that might be indicative of the potential for a future intrusion or the occurrence of a past intrusion of an IT System or events that are indicative of inappropriate activity that may have resulted from misuse, access, or malicious activity of IT System assets must be collected.
P.ANALYZ	Analytical processes and information to derive conclusions about intrusions (past, present, or future) must be applied to IDS data and appropriate response actions taken.
P.MANAGE	The TOE shall only be managed by authorized users.

P.ACCESS	All data collected and produced by the TOE shall only be used for authorized purposes.
P.ACCACT	Users of the TOE shall be accountable for their actions within the IDS.
P.INTGTY	Data collected and produced by the TOE shall be protected from modification.
P.PROTCT	The TOE shall be protected from unauthorized accesses and disruptions of TOE data and functions.

4. Security Objectives

This section identifies the security objectives of the TOE and its supporting environment. The security objectives identify the responsibilities of the TOE and its environment in meeting the security needs. They were copied verbatim from the security environment as described in the *U.S. Government Protection Profile Intrusion Detection System System For Basic Robustness Environments,* to which this ST claims conformance.

4.1 Information Technology (IT) Security Objectives

The following are the TOE security objectives:

O.PROTCT	The TOE must protect itself from unauthorized modifications and access to its functions and data.
O.IDSCAN	The Scanner must collect and store static configuration information that might be indicative of the potential for a future intrusion or the occurrence of a past intrusion of an IT System.
O.IDSENS	The Sensor must collect and store information about all events that are indicative of inappropriate activity that may have resulted from misuse, access, or malicious activity of IT System assets and the IDS.
O.IDANLZ	The Analyzer must accept data from IDS Sensors or IDS Scanners and then apply analytical processes and information to derive conclusions about intrusions (past, present, or future).
O.RESPON	The TOE must respond appropriately to analytical conclusions.
O.EADMIN	The TOE must include a set of functions that allow effective management of its functions and data.
O.ACCESS	The TOE must allow authorized users to access only appropriate TOE functions and data.
O.IDAUTH	The TOE must be able to identify and authenticate users prior to allowing access to TOE functions and data.
O.OFLOWS	The TOE must appropriately handle potential audit and System data storage overflows.
O.AUDITS	The TOE must record audit records for data accesses and use of the System functions.
O.AUDIT_SORT	The TOE will provide the capability to sort the audit information.
O.INTEGR	The TOE must ensure the integrity of all audit and System data.
O.TRAFFIC	The TOE must provide a capability to filter network traffic based on combinations of protocol, IP address and port.

4.2 Security Objectives for the Environment

The TOE's operating environment must satisfy the following objectives.

OE.AUDIT_PROTECTION	The IT Environment will provide the capability to protect audit information.
OE.TIME	The IT Environment will provide reliable timestamps to the TOE.
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 access credentials 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 is interoperable with the IT System it monitors.
OE.CONFID	The IT Environment will provide the capability to protect the confidentiality of data communicated by the administrative users to the TOE.

5. IT Security Requirements

Most of the security requirements for the TOE have been drawn from Parts 2 and 3 of the Common Criteria as well as from the *U.S. Government Protection Profile Intrusion Detection System System for Basic Robustness Environments Protection Profile.* The security functional requirements have been selected to correspond to the actual security functions implemented by the TOE and as required by the PP, while the assurance requirements have been selected to offer assurance that those security functions are properly realized.

This security target utilizes extended requirements only as reproductions of requirements found in the protection profile to which this security target is claiming compliance. Therefore, all requirements for information related to the extended requirements are satisfied by this security target's compliance with validated protection profiles.

5.1 Extended Components Definition

The *U.S. Government Protection Profile Intrusion Detection System for Basic Robustness Environments* (IDSSPP) defines extended security functional requirements, which are included in this ST. The IDSSPP provides a rationale for the use of extended security requirements, identifying that the CC audit family (FAU) was used as a model.

5.2 TOE Security Functional Requirements

This section specifies the security functional requirements (SFRs) for the TOE. All SFRs were drawn from Part 2 of the Common Criteria v3.1 Revision 2 and the Protection Profile (PP) identified in the Protection Profile Claims section.

This ST includes a number of extended requirements. Each of the extended requirements is defined in the *U.S. Government Protection Profile Intrusion Detection System For Basic Robustness Environments.* The extended requirements can be identified by the use of the keyword "EXT" in the title.

Every SFR included in the PP is addressed in this ST. Note, however, that the PP was written using CC v3.1, Revision 1, whereas this ST claims conformance to CC v3.1, Revision 2. The SFRs therefore reproduce the wording of CC Part 2 v3.1, Revision 2. Section 7 (Protection Profile Claims) identifies the SFRs whose wording, for this reason, differs from that used in the PP.

The following table describes the SFRs that are satisfied by LogRhythm.

Requirement Class	Requirement Component
FAU: Security Audit	FAU_GEN.1: Audit Data Generation
	FAU_SAR.1: Audit review
	FAU_SAR.2: Restricted audit review
	FAU_SAR.3: Selectable audit review
	FAU_SEL.1: Selective Audit
	FAU_STG.2: Guarantee of audit data availability
	FAU_STG.4: Prevention of audit data loss
FIA: Identification and Authentication	FIA_UAU.2: User authentication before any action
	FIA_ATD.1: User attribute definition
	FIA_UID.2: User identification before any action
FMT: Security Management	FMT_MOF.1: Management of security functions
	FMT_MTD.1: Management of TSF data
	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
IDS Component Requirements (IDS)	IDS_ANL.1: Analyzer analysis
	IDS_RCT.1: Analyzer react

Requirement Class	Requirement Component
	IDS_RDR.1: Restricted data review
	IDS_STG.1(1): Guarantee of analyzer data availability – General IDS_STG.1(2): Guarantee of analyzer data availability – AI Engine Server
	IDS_STG.2(1): Prevention of Analyzer data loss – General IDS_STG.2(2): Prevention of Analyzer data loss – AI Engine Server
	IDS_SDC.1: System Data Collection

Table 1 - TOE Security Functional Components

5.2.1 Security Audit (FAU)

5.2.1.1 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 [***basic***] level of audit; and

c) [**Access to the System and access to the TOE and System 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 (if applicable), and the outcome (success or failure) of the event; and

b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, [**the additional information specified in the Details column of Table 2 Auditable Events**].

Component	Event	Details
FAU_GEN.1	Start-up and shutdown of audit functions	
FAU_GEN.1	Access to System	
FAU_GEN.1	Access to the TOE and System data	**Object IDs, Requested access**
FAU_SAR.1	Reading of information from the audit records	
FAU_SAR.2	Unsuccessful attempts to read information from the audit records	
FAU_SEL.1	All modifications to the audit configuration that occur while the audit collection functions are operating	
FIA_UAU.2	All use of the authentication mechanism	**User identity, location**
FIA_UID.2	All use of the user identification mechanism	**User identity, location**
FMT_MOF.1	All modifications in the behavior of the functions of the TSF	
FMT_SMF.1	Use of the management functions.	
FMT_MTD.1	All modifications to the values of TSF data	
FMT_SMR.1	Modifications to the group of users that are part of a role	**User identity**

Table 2 - Auditable Events

5.2.1.2 Audit review (FAU_SAR.1)

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

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

5.2.1.3 Restricted audit review

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

5.2.1.4 Selectable audit review

FAU_SAR.3.1 The TSF shall provide the ability to apply [**sorting**] of audit data based on [**date and time, subject identity, type of event, and success or failure of related event**].

5.2.1.5 Selective audit (FAU_SEL.1)

FAU_SEL.1.1 The TSF shall be able to include or exclude auditable events from the set of audited events based on the following attributes:
a) [***event type***];
b) [**no additional attributes**].

5.2.1.6 Guarantees of Audit Data Availability (FAU_STG.2)

FAU_STG.2.1 The TSF shall protect the stored audit records in the audit trail from unauthorized deletion.

FAU_STG.2.2 The TSF shall be able to [***prevent***] **unauthorized** modifications to the stored audit records in the audit trail.

FAU_STG.2.3 The TSF shall ensure that [**100% of existing**] audit records will be maintained when the following conditions occur: [***audit storage exhaustion***].

5.2.1.7 Prevention of audit data loss

FAU_STG.4.1 The TSF shall [***prevent auditable events, ~~except those taken by the authorised user with special rights~~***] and [**send an alarm**] if the audit trail is full.

5.2.2 Identification and Authentication (FIA)

5.2.2.1 User Attribute Definition (FIA_ATD.11)

FIA_ATD.1.1 The TSF shall maintain the following list of security attributes belonging to individual users: [
a) User identity;
b) Authentication data;
c) Authorisations; and
d) [*no other security attributes*]].

5.2.2.2 User Identification Before Any Action (FIA_UID.2)

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

5.2.2.3 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.

5.2.3 Security Management (FMT)

5.2.3.1 Management of Security Functions Behaviour (FMT_MOF.1)

FMT_MOF.1.1 The TSF shall restrict the ability to [***modify the behaviour of***] the functions [**of System data collection, analysis and reaction**] to [**authorised System administrators**].

5.2.3.2 Management of TSF Data (FMT_MTD.1)

FMT_MTD.1.1 The TSF shall restrict the ability to [***query* and add System and audit data, and shall restrict the ability to query and modify all other TOE data**] to [**authorized System administrators]**.

Application Note: The statement "query and add System and audit data" in this requirement refers to the ability to look at and to change the set of events for which audit and System log records are actually collected. It does not refer to the capability of looking at and changing the data in these logs after it has been collected. The ability to look at the records within the audit log is specified using FAU_SAR.1. The ability to look at the records within the System data log is specified using IDS_RDR.1. Furthermore, FMT_MTD.1 is included to satisfy a dependency of FAU_SEL.1. In order to properly satisfy this dependency, FMT_MTD.1 needs to address management of the collection of audit data.

5.2.3.3 Specification of Management Functions (FMT_SMF.1)

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

a.) **Management of user accounts**

b.) **Management of audit data and audit configurations**

c.) **Management of System Data collection, analysis and reaction**].

5.2.3.4 Security Roles (FMT_SMR.1)

FMT_SMR.1.1 The TSF shall maintain the following *roles:* [**authorised administrator, authorised System administrators, and [authorized Restricted Analyst administrators]**].

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

Application Note: LogRhythm has two pre-defined administrative user roles: Global Admin and Global Analyst. LogRhythm provides user profiles for defining Restricted Analyst role. The Global Admin role corresponds with the PP role: 'System administrator'. The Global Analyst corresponds with the PP role: 'administrator'. Restricted Analyst role is a ST-defined role.

5.2.4 Protection of the TSF (FPT)

5.2.4.1 Basic Internal TSF Data Transfer Protection (FPT_ITT.1)

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

5.2.5 IDS Component Requirements (IDS)

5.2.5.1 IDS_SDC.1 System Data Collection (EXT)

IDS_SDC.1.1 The System shall be able to collect the following information from the targeted IT System resource(s):

a) [***Start-up and shutdown, identification and authentication events, data accesses, service requests, network traffic, security configuration changes, data introduction, detected malicious code, access control configuration, service configuration, authentication configuration., accountability policy configuration, detected known vulnerabilities***]; and

b) [**none**]. (EXT)

IDS_SDC.1.2 At a minimum, the System shall collect and record the following information:

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

b) The additional information specified in the Details column of Table 3 System Events. (EXT)

Component	Event	Details
IDS_SDC.1	Start-up and shutdown	**none**
IDS_SDC.1	Identification and authentication events	**User identity, location, source address, destination address**
IDS_SDC.1	Data accesses	**Object IDS, requested access, source address, destination address**
IDS_SDC.1	Service Requests	**Specific service, source address, destination address**
IDS_SDC.1	Network traffic	**Protocol, source address, destination address**
IDS_SDC.1	Security configuration changes	**Source address, destination address**
IDS_SDC.1	Data introduction	**Object IDS, location of object, source address, destination address**
IDS_SDC.1	Start-up and shutdown of audit functions	**none**
IDS_SDC.1	Detected malicious code	**Location, identification of code**
IDS_SDC.1	Access control configuration	**Location, access settings**
IDS_SDC.1	Service configuration	**Service identification (name or port), interface, protocols**
IDS_SDC.1	Authentication configuration	**Account names for cracked passwords, account policy parameters**
IDS_SDC.1	Accountability policy configuration	**Accountability policy configuration parameters**
IDS_SDC.1	Detected known vulnerabilities	**Identification of the known vulnerability**

Table 3 - System Events

5.2.5.2 IDS_ANL.1 Analyser analysis (EXT)

IDS_ANL.1.1 The System shall perform the following analysis function(s) on all IDS data received:

a) [***signature, integrity***]; and

b) [**none**]. (EXT)

IDS_ANL.1.2 The System shall record within each analytical result at least the following information:
a) Date and time of the result, type of result, identification of data source; and
b) [**none**]. (EXT)

5.2.5.3 IDS_RCT.1 Analyser react (EXT)

IDS_RCT.1.1 The System shall send an alarm to [**the alarm database and users configured to receive alarms**] and take [**configured remediation actions**] when an intrusion is detected. (EXT)

5.2.5.4 IDS_RDR.1 Restricted Data Review (EXT)

IDS_RDR.1.1 The System shall provide [**authorized administrators, authorized System Administrators, authorized Restricted Analyst Administrators**] with the capability to read [**all analyzer data that the administrator is authorized to view**] from the System data. (EXT)

IDS_RDR.1.2 The System shall provide the System data in a manner suitable for the user to interpret the information. (EXT)

IDS_RDR.1.3 The System shall prohibit all users read access to the System data, except those users that have been granted explicit read-access. (EXT)

5.2.5.5 IDS_STG.1(1) Guarantee of System Data Availability – General

IDS_STG.1.1(1) The System shall protect the stored System data from unauthorized deletion. (EXT)

IDS_STG.1.2(1) The System shall protect the stored System data from modification. (EXT)

IDS_STG.1.3(1) The **System Monitor Agent, Log Manager, and Event Manager components of the** System shall ensure that [**100% of existing**] System data will be maintained when the following conditions occur: [***System data storage exhaustion***]. (EXT)

Application Note: The TOE is designed as a distributed system, with AI Engine Server as an optional component. This ST iterates the System data storage requirements to specify behavior applicable to each component.

5.2.5.6 IDS_STG.1(2) Guarantee of System Data Availability – AI Engine Server

IDS_STG.1.1(2) The System shall protect the stored System data from unauthorized deletion. (EXT)

IDS_STG.1.2(2) The System shall protect the stored System data from modification. (EXT)

IDS_STG.1.3(2) The **AI Engine Server component of the** System shall ensure that [**a block with System administrator-configured size of**] System data will be maintained when the following conditions occur: [***System data storage exhaustion***]. (EXT)

5.2.5.7 IDS_STG.2(1) Prevention of System data loss (EXT) – General

IDS_STG.2.1(1) The **System Monitor Agent, Log Manager, and Event Manager components of the** System shall [***ignore System data***] and send an alarm if the storage capacity has been reached. (EXT)

5.2.5.8 IDS_STG.2(2) Prevention of System data loss (EXT) – AI Engine Server

IDS_STG.2.1(2) The **AI Engine Server component of the** System shall [***overwrite the oldest stored System data***] and send an alarm if the storage capacity has been reached. (EXT)

5.3 TOE Security Assurance Requirements

The security assurance requirements for the TOE are the EAL 2 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
ADV: Development	ADV_ARC.1: Security architecture description

Requirement Class	Requirement Component
	ADV_FSP.2: Security-enforcing functional specification
	ADV_TDS.1: Basic design
AGD: Guidance documents	AGD_OPE.1: Operational user guidance
	AGD_PRE.1: Preparative procedures
ALC: Life-cycle support	ALC_CMC.2: Use of a CM system
	ALC_CMS.2: Parts of the TOE CM coverage
	ALC_DEL.1: Delivery procedures
	ALC_FLR.2: Flaw reporting procedures
ATE: Tests	ATE_COV.1: Evidence of coverage
	ATE_FUN.1: Functional testing
	ATE_IND.2: Independent testing - sample
AVA: Vulnerability assessment	AVA_VAN.2: Vulnerability analysis

Table 4 - EAL 2 augmented with ALC_FLR.2 Assurance Components

5.3.1 Development (ADV)

5.3.1.1 Security Architecture Description (ADV_ARC.1)

ADV_ARC.1.1d The developer shall design and implement the TOE so that the security features of the TSF cannot be bypassed.
ADV_ARC.1.2d The developer shall design and implement the TSF so that it is able to protect itself from tampering by untrusted active entities.
ADV_ARC.1.3d The developer shall provide a security architecture description of the TSF.
ADV_ARC.1.1c The security architecture description shall be at a level of detail commensurate with the description of the SFR-enforcing abstractions described in the TOE design document.
ADV_ARC.1.2c The security architecture description shall describe the security domains maintained by the TSF consistently with the SFRs.
ADV_ARC.1.3c The security architecture description shall describe how the TSF initialization process is secure.
ADV_ARC.1.4c The security architecture description shall demonstrate that the TSF protects itself from tampering.
ADV_ARC.1.5c The security architecture description shall demonstrate that the TSF prevents bypass of the SFR-enforcing functionality.
ADV_ARC.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.1.2 Security-enforcing Functional Specification (ADV_FSP.2)

ADV_FSP.2.1d The developer shall provide a functional specification.
ADV_FSP.2.2d The developer shall provide a tracing from the functional specification to the SFRs.
ADV_FSP.2.1c The functional specification shall completely represent the TSF.
ADV_FSP.2.2c The functional specification shall describe the purpose and method of use for all TSFI.
ADV_FSP.2.3c The functional specification shall identify and describe all parameters associated with each TSFI.
ADV_FSP.2.4c For SFR-enforcing TSFIs, the functional specification shall describe the SFR-enforcing actions associated with the TSFI.
ADV_FSP.2.5c For SFR-enforcing TSFIs, the functional specification shall describe direct error messages resulting from processing associated with the SFR-enforcing actions.
ADV_FSP.2.6c The tracing shall demonstrate that the SFRs trace to TSFIs in the functional specification.
ADV_FSP.2.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADV_FSP.2.2E The evaluator shall determine that the functional specification is an accurate and complete instantiation of the SFRs.

5.3.1.3 Basic Modular Design (ADV_TDS.1)

ADV_TDS.1.1d The developer shall provide the design of the TOE.
ADV_TDS.1.2d The developer shall provide a mapping from the TSFI of the functional specification to the lowest level of decomposition available in the TOE design.
ADV_TDS.1.1c The design shall describe the structure of the TOE in terms of subsystems.
ADV_TDS.1.2c The design shall identify all subsystems of the TSF.
ADV_TDS.1.3c The design shall describe the behavior of each SFR-supporting or SFR-non-interfering TSF subsystem in sufficient detail to determine that it is not SFR-enforcing.
ADV_TDS.1.4c The design shall summarize the SFR-enforcing behavior of the SFR-enforcing subsystems.
ADV_TDS.1.5c The design shall provide a description of the interactions among SFR-enforcing subsystems of the TSF, and between the SFR-enforcing subsystems of the TSF and other subsystems of the TSF.
ADV_TDS.1.6c The mapping shall demonstrate that all behavior described in the TOE design is mapped to the TSFIs that invoke it.
ADV_TDS.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
ADV_TDS.1.2e The evaluator shall determine that the design is an accurate and complete instantiation of all security functional requirements.

5.3.2 Guidance Documents (AGD)

5.3.2.1 Operational User Guidance (AGD_OPE.1)

AGD_OPE.1.1d The developer shall provide operational user guidance.
AGD_OPE.1.1c The operational user guidance shall describe, for each user role, the user-accessible functions and privileges that should be controlled in a secure processing environment, including appropriate warnings.
AGD_OPE.1.2c The operational user guidance shall describe, for each user role, how to use the available interfaces provided by the TOE in a secure manner.
AGD_OPE.1.3c The operational user guidance shall describe, for each user role, the available functions and interfaces, in particular all security parameters under the control of the user, indicating secure values as appropriate.
AGD_OPE.1.4c The operational user guidance shall, for each user role, clearly present each type of security-relevant event relative to the user-accessible functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.
AGD_OPE.1.5c The operational user guidance 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.
AGD_OPE.1.6c The operational user guidance shall, for each user role, describe the security measures to be followed in order to fulfill the security objectives for the operational environment as described in the ST.
AGD_OPE.1.7c The operational user guidance shall be clear and reasonable.
AGD_OPE.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.2.2 Preparative Procedures (AGD_PRE.1)

AGD_PRE.1.1d The developer shall provide the TOE including its preparative procedures.
AGD_PRE.1.1c The preparative procedures shall describe all the steps necessary for secure acceptance of the delivered TOE in accordance with the developer's delivery procedures.
AGD_PRE.1.2c The preparative procedures shall describe all the steps necessary for secure installation of the TOE and for the secure preparation of the operational environment in accordance with the security objectives for the operational environment as described in the ST.

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

AGD_PRE.1.2e The evaluator shall apply the preparative procedures to confirm that the TOE can be prepared securely for operation.

5.3.3 Life-cycle Support (ALC)

5.3.3.1 Use of a CM System (ALC_CMC.2)

ALC_CMC.2.1d The developer shall provide the TOE and a reference for the TOE.

ALC_CMC.2.2d The developer shall provide the CM documentation.

ALC_CMC.2.3d The developer shall use a CM system.

ALC_CMC.2.1c The TOE shall be labeled with its unique reference.

ALC_CMC.2.2c The CM documentation shall describe the method used to uniquely identify the configuration items.

ALC_CMC.2.3c The CM system shall uniquely identify all configuration items.

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

5.3.3.2 Parts of the TOE CM Coverage (ALC_CMS.2)

ALC_CMS.2.1d The developer shall provide a configuration list for the TOE.

ALC_CMS.2.1c The configuration list shall include the following: the TOE itself; the evaluation evidence required by the SARs; and the parts that comprise the TOE.

ALC_CMS.2.2c The configuration list shall uniquely identify the configuration items.

ALC_CMS.2.3c For each TSF relevant configuration item, the configuration list shall indicate the developer of the item.

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

5.3.3.3 Delivery Procedures (ALC_DEL.1)

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

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

ALC_DEL.1.1c The delivery documentation shall describe all procedures that are necessary to maintain security when distributing versions of the TOE to the consumer.

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

5.3.3.4 Flaw Reporting Procedures (ALC_FLR.2)

ALC_FLR.2.1d The developer shall document 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 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 remediated and the remediation procedures 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.

5.3.4 Tests (ATE)

5.3.4.1 Analysis of Coverage (ATE_COV.1)

ATE_COV.1.1d The developer shall provide evidence of the test coverage.
ATE_COV.1.1c The evidence of the test coverage shall show the correspondence between the tests in the test documentation and the TSFIs in the functional specification.
ATE_COV.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.4.2 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, expected test results and actual test results.
ATE_FUN.1.2c The test plans shall identify the tests to be performed and describe the scenarios for performing each test. These scenarios shall include any ordering dependencies on the results of other tests.
ATE_FUN.1.3c The expected test results shall show the anticipated outputs from a successful execution of the tests.
ATE_FUN.1.4c The actual test results shall be consistent with the expected test results.
ATE_FUN.1.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.4.3 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 execute a sample of tests in the test documentation to verify the developer test results.
ATE_IND.2.3e The evaluator shall test a subset of the TSF to confirm that the TSF operates as specified.

5.3.5 Vulnerability Assessment (AVA)

5.3.5.1 Vulnerability Analysis (AVA_VAN.2)

AVA_VAN.2.1d The developer shall provide the TOE for testing.
AVA_VAN.2.1c The TOE shall be suitable for testing.
AVA_VAN.2.1e The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
AVA_VAN.2.2e The evaluator shall perform a search of public domain sources to identify potential vulnerabilities in the TOE.
AVA_VAN.2.3e The evaluator shall perform an independent vulnerability analysis of the TOE using the guidance documentation, functional specification, TOE design, and security architecture description to identify potential vulnerabilities in the TOE.

AVA_VAN.2.4e The evaluator shall conduct penetration testing, based on the identified potential vulnerabilities, to determine that the TOE is resistant to attacks performed by an attacker possessing Basic attack potential.

6. TOE Summary Specification

This chapter describes the TOE security functions.

6.1 TOE Security Functions

6.1.1 Security Audit

The TOE has an audit generation mechanism to record security events at a basic level of audit.

The TOE implements auditing functionality compliant with the IDSSPP. Some of the events which can be audited include the following: startup and shutdown of the TOE's auditing function; successful and unsuccessful attempts to read the audit records; access to the TOE, the log records collected by the TOE, and events identified by the TOE; all use of identification and authentication mechanisms; modifications in the behavior of the TOE security functions; modifications to the values of TSF data; and modifications to a user's security management role (FAU_GEN.1.1). A complete list follows. The TOE records the following data for each audited event: the date and time of the event; the type of event, the subject identity (if applicable); the outcome of the event; and other information specific to the event types such as a user's location and identity for identification and authentication attempts. (see FAU_GEN.1.2). The TOE provides stored procedures for the authorized System administrator to read audit records from the audit trail. An authorized System administrator executes the stored procedures in SQL Server Management Studio, a component of SQL Server. (FAU_SAR.1) and this interface is restricted to the authorized System administrator role (FAU_SAR.2). The audit stored procedure provides flexible sorting and filtering, including the capability to sort audit records on the following fields in the audit data: date and time; subject identity; type of event; and success or failure of related event. (FAU_SAR.3). The TOE also provides the functionality to include or exclude (turn on or off) auditable events from the set of audited events based on "event type" (FAU_SEL.1). Finally, the TOE prevents unauthorized modifications and deletions to the stored audit records. There are no TOE interfaces provided to modify or delete stored audit records. The audit records reside in SQL Server trace files in host file systems. The TOE minimizes the loss of audit data in the event the space available for storing audit records is exhausted A stored procedure defines the SQL Server trace such that the SQL Server instance shuts down when storage is exhausted. This maintains 100% of existing audit records and no more audit data will be written to the trace file. SQL Server shut down also generates a Windows event, which serves as an alarm that security audit storage is exhausted. (FAU_STG.2, FAU_STG.4 (TOE)).

The TOE is a software only implementation and therefore relies on the operational environment to provide a reliable timestamp. The audit records are stored in tables in a SQL Server 2008 R2 database, which ultimately are stored on a Windows Server 2008 R2 file system. Therefore the TOE relies on the operating environment for proper enforcement of file permission settings.

The following basic events which the TOE can audit are:

1. Start-up and shutdown of audit functions
2. Access to System
3. Access to the TOE System data
4. Reading of information from the audit records
5. Unsuccessful attempts to read information from the audit records
6. All modifications to the audit configuration that occur while the audit collection functions are operating
7. All use of the authentication mechanism
8. All use of the user identification mechanism
9. All modifications in the behaviour of the functions of the TSF
10. All modifications to the values of TSF data
11. Modifications to the group of users that are part of a role

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 audit function, all authentication attempts, all administrative actions, and all required auditable events as specified in Table 2. At a minimum, each event includes date and time, logging level (event type), subject identity, and outcome of event.
- FAU_SAR.1: The TOE provides System administrators with the ability to read the audit records. These records are provided in a humanly readable format to enable the reader to interpret the information.
- FAU_SAR.2: The TOE restricts read access to the audit logs to only those users given explicit read-access.
- FAU_SAR.3: The TOE provides System administrators with the ability to sort the audit records based on date and time, subject identity, type of event, and success or failure of related event.
- FAU_SEL.1: The TOE provides System administrators with the capability to include or exclude audit events based on event type.
- FAU_STG.2: The TSF protects the stored audit records in the audit trail from unauthorized deletion, prevents unauthorized modifications to the stored audit records in the audit trail, and ensures that 100% of existing audit records is preserved when audit storage is exhausted.
- FAU_STG.4: The TSF prevents auditable events and sends an alarm when the audit trail is full.

6.1.2 Identification and Authentication

Users and roles are defined in the TOE, operating at the application layer (FIA_ATD.1). In the evaluated configuration, a user's authentication data consists of their identity in the local Windows OS or in Active Directory. The User identity and authorizations (which includes the user's role) are added to an internal SCUSER table and then propagated to SQL Server.

When an authorized System administrator creates a login for a TOE user, the user must first have an existing "Person" record defined in the TOE, which consists of a username and contact method (i.e. email) for alarming purposes. Once the administrator creates the person record, he can link the person to an existing Windows login (local OS or Active Directory). The sequence of events is as follows:

1. The LogRhythm Console connects to the Event Manager SQL Server and creates a SQL Server login using the supplied local Windows OS or Active Directory user account.
2. The Event Manager SQL Server stores the login in its internal tables.
3. The LogRhythm Console extracts the SQL Server login from the Event Manager SQL Server and stores it in a LogRhythm database table
4. The LogRhythm Console then adds the new login as a database user to all the Event Manager databases with the appropriate database role membership based on the LogRhythm user's role (Global Admin, Global Analyst, or Restricted Analyst).
5. The LogRhythm Console then adds the login to each Log Manager the user has been granted access to. The Log Manager SQL Server stores the login in its internal tables.
6. The LogRhythm Console also adds the new Log Manager SQL Server login as a database user to the Log Manager databases with the appropriate database role membership based on the LogRhythm user's role (Global Admin, Global Analyst, or Restricted Analyst).

When in the evaluated configuration, SQL Server authentication is disabled and authentication is performed by the local Windows OS or by Active Directory. The TOE supports Common Access Cards (CAC) for user credentials as well as local Windows OS or Active Directory username/password credentials.

When logging in to the Console, the user specifies their user identity and login with their Windows OS, Active Directory, or CAC user account. If the login fails then access to the TOE is denied. If login is successful, then the application table is checked for the user's authorizations. Each user authorization consists of their assigned role (all administrative users must have an assigned role) and in the case of users possessing a Restricted Analyst role the user will also have a list of log sources that the user was granted access to by the authorized System administrator.

If the user's identity is not in the table, then access is denied. In the evaluated configuration, the TOE stores Windows account information but not user credentials. When a user logs in to the TOE, the local Windows OS or Active Directory authenticates the claimed user identity (FIA_UAU.2, FIA_UID.2). If successful, the application table is checked for the user's identity. If the user identity is not in the table then access is denied.

If a user has an Active Directory account but it is not administratively granted a logon to the Console, then they will be prevented from logging in using their Active Directory account.

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

- FIA_ATD.1: The TOE maintains user information. The following information is associated with each administrator account: username, role, and Windows user account. In addition in the case of users possessing the Restricted Analyst Administrator role, 'authorizations' are also associated with the user. These authorizations represent the log sources that the user was granted access to by the authorized administrator.
- FIA_UID.2: The TOE requires each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.
- FIA_UAU.2: The TOE requires the user to successfully authenticate by entering their identity and Windows user account credentials. The information is provided to the operational environment where if the user is authenticated and their userid is subsequently found in the application table then the user is permitted access to the TOE. If the user is not successfully authenticated or if userid is not found in the application table then access is denied.

6.1.3 Security Management

The TOE provides the capability to manage the auditing, analysis and reaction functions. The management functions are restricted to the authorized System administrator role.

The TOE comes with two pre-defined roles: Global Admin and Global Analyst. The TOE provides user profiles for a customer to define Restricted Analyst role. These roles, when assigned to users, provide varying levels of access to the TOE interfaces and functions (FMT_SMR.1).

The Global Admin role has full control of the configuration and data. This role coincides with the access level defined in the PP of Authorized System administrator, which is defined as a user having the ability to configure and access the TOE users and data, and modify the behavior of the analysis and reaction functions (FMT_MOF.1). This includes configuration of alarm rules, and archive policies and settings. The Global Admin has the overall responsibility of managing and configuring the TOE. The Global Admin can create, modify, delete, configure, download updates and implement the rules on the TOE. In addition, the Administrator is also the only role that can manage the security settings on the system, such as user accounts and audit settings, and restore inactive archived files.

The Global Admin can grant users with the Restricted Analyst role access to database resources pertaining to specific Log Sources. These permissions (list of log sources) are stored in the system table as part of the user's authorizations. By default, a newly created user with Restricted Analyst selected as their User Role is not assigned access permission to any Log Sources.

The Global Analyst role coincides with the access level defined in the PP of Authorized Administrator. Users with the Global Analyst role have read-only access to all System data. In addition, the Global Analyst Administrator can create customized rules, but they do not have the authority to enable the rules; only the System Administrator can enable the rules.

Users must have one of the three administrative roles assigned to them in order to access any of the TOE security functions.

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

- FMT_MOF.1: The TOE provides and restricts the capability to manage the analysis and reaction functions as identified in FMT_MOF.1.
- FMT_MTD.1: The TOE provides and restricts the capability to manage the selection of audit events.

- FMT_SMF.1: The TOE provides interfaces to manage the audit generation function.
- FMT_SMR.1: The TOE maintains user role attributes. There are three pre-defined built-in roles.

6.1.4 Protection of the TSF

All communication channels between TOE components are protected by TLS based on FIPS 140-2 certified cryptographic modules (FPT_ITT.1).

Data sent between the Console, Mediator Server Service, AI Engine Service, ARM, and Job Manager and SQL Server 2008 R2 databases uses the SQL TDS (Tabular Data Stream) protocol. TDS is used with the Windows TLS/SSL Security Support Provider for TLS communication between SQL clients (e.g., Console, Mediator Server, AI Engine, ARM, and Job Manager) and SQL Server. Data sent between System Monitor Agents and Mediator Servers and between Mediator Servers and AI Engine Communication Managers are protected using TLS services as described below.

The TOE has configurations to support both one-way and two-way authentication in TLS communication. In the one-way authentication configuration, each Mediator Server service, each AI Engine Communication Manager, and each SQL Server generates a self-signed server certificate. TLS clients use these certificates to authenticate the identity of the servers. That is, a System Monitor agent authenticates its Mediator Server with the Mediator's server certificate, a Mediator Server authenticates its AI Engine Server with the AI Engine Server's server certificate, and all the SQL clients (such as Consoles) authenticate their SQL servers with the server's certificate. A Mediator Server service will accept connections only from registered System Monitor Agents. Similarly, an AI Engine Service will accept connections only from a list of authorized Mediator Servers. In this configuration, TLS clients do not validate server certificates and TLS servers do not require client certificates.

In two-way authentication configurations, the TOE supports both self-signed and System administrator-specified server certificates. A System administrator can specify a Mediator Server Service server certificate, an AI Engine Service server certificate, or both. If the System administrator does not specify a server certificate, the TOE defaults to a self-signed certificate as in the one-way authentication configuration. A System administrator can specify a SQL Server TLS server certificate through the operational environment. Moreover, a System administrator can configure each Mediator Server Service and each AI Engine Communication Manager to require a TLS client certificate. The System administrator also specifies which TLS client certificates to use when a TLS server requires client certificates. Both TLS clients and servers can be configured to validate certificates (host identity, trusted authority check) and to check certificate revocation. As with the one-way authentication configuration, a Mediator Server service will accept connections only from a registered System Monitor Agents and an AI Engine Service will accept connections only from a list of authorized Mediator Servers.

The following communication channels are protected:

- Console to/from SQL Server 2008 R2 Server (Event Manager or Log Manager) Communications

  Encryption is provided when 'Encrypt all communications' on the Console Login Screen has been configured. Once the Console starts up, the Console reads the Server field from the console logon screen and initiates a connection to the SQL Server. All connections and communications are handled by SQL Server's SSL and encryption protocols.

  SQL Server module: Windows Server 2008 R2 Enhanced Cryptographic Provider (RSAENH)
  FIPS 140-2 certificate: #1337

  Console module: LogRhythm 6.0.4 Console
  FIPS 140-2 certificate: #1807

- Log Manager, AI Engine Server, and Event Manager to/from SQL Server 2008 R2 (Event Manager):

  Each SQL client (Mediator Server (Log Manager), AI Engine (AI Engine Server), ARM (Event Manager) and Job Manager (Event Manager)) uses SQL Server TDS with TLS to establish a connection to the SQL Server on the Event Manager SQL database named LogRhythmEMDB. Once the Mediator Server starts up, it reads the scmedsvr.ini file found in the LogRhythm Mediator Server\config file, obtains the IP Address of the SQL Server, and then initiates a connection to the SQL Server. Similarly, the AI Engine, ARM, and Job Manager obtain the SQL Server IP address from

their initialization files and initiate connections to the SQL Server on the Event Manager. All connections and communications are handled by SQL Server's SSL and encryption protocols. Data sent between the Mediator Server, AI Engine, ARM, and Job Manager and SQL Server database uses the SQL TDS protocol. TDS is used with the Windows TLS/SSL Security Support Provider between SQL clients and SQL Server. All traffic between LogRhythm software (Mediator Server, AI Engine, ARM, Job Manager, and Console) and SQL Server is encrypted.

SQL Server module: Windows Server 2008 R2 Enhanced Cryptographic Provider (RSAENH)
FIPS 140-2 certificate: #1337

Mediator Server module: LogRhythm 6.0.4 Log Manager
FIPS 140-2 certificate: #1808

AI Engine module: LogRhythm 6.0.4 AI Engine Server
FIPS 140-2 certificate: #1805

ARM module: LogRhythm 6.0.4 Event Manager
FIPS 140-2 certificate: #1817

Job Manager module: LogRhythm 6.0.4 Event Manager
FIPS 140-2 certificate: #1817

- System Monitor Agent to Log Manager Communications:

  Once the System Monitor Agent starts up, it reads the scsm.ini file found in the LogRhythm System Monitor Agent config file to obtain the IP Address of the Mediator. The System Monitor Agent then initiates a connection to the Mediator Server. A Mediator Server will accept a connection only from a registered System Monitor Agent. Registration establishes the IP address of each System Monitor Agent. The System Monitor Agent and Mediator Server negotiate a secure communication channel as configured (for example with one-way or two-way authentication). All traffic between System Monitor Agents and the Mediator Server including log data and System Monitor Agent configuration/management data is encrypted.

  Mediator Server module: LogRhythm 6.0.4 Log Manager
  FIPS 140-2 certificate: #1808

  Windows System Monitor Agent module: LogRhythm 6.0.4 Windows System Monitor Agent
  FIPS 140-2 certificate: #1806

  UNIX System Monitor Agent module: OpenSSL FIPS Object Module (Software Version: 1.2.3)
  FIPS 140-2 certificate: #1051

  UNIX System Monitor Agent module: OpenSSL FIPS Runtime Module (Software Version 1.2)
  FIPS 140-2 certificate: #1111

- Log Manager to AI Engine Communication Manager:

  When a Mediator Server starts up, it reads its initialization file to obtain the IP Address of the AI Engine Communications Manager. The Mediator Server then initiates a connection to the AI Engine Communication Manager. An AI Engine Communication Manager will accept connections only from a list of authorized Mediator Servers. The Mediator Server and AI Engine Communication Manager negotiate a secure communication channel as configured (for example, with one-way or two-way authentication). All traffic between Mediator Servers and the Communication Manager is encrypted.

  Mediator Server module: LogRhythm 6.0.4 Log Manager
  FIPS 140-2 certificate: #1808

  AI Engine Communication Manager module: LogRhythm 6.0.4 AI Engine Server
  FIPS 140-2 certificate: #1805

The Mediator Server, AI Engine, ARM, Console, and Job Manager are SQL clients that use a User ID and password to authenticate into SQL Server using SQL Server security. The Mediator Server, AI Engine, ARM, and Job

Manager keep these User IDs and passwords in their initialization files in their configuration directories. The Console prompts for these credentials on startup. The Mediator Server, AI Engine, ARM, and Job Manager can all be configured to use Windows or Active Directory credentials for the services to avoid having user names and passwords in the clear in their respective initialization files. Windows authentication must be used in the evaluated configuration. The Windows Service Account for each SQL client above must also be granted access to the appropriate databases in SQL Server.

The integrity of LogRhythm archives is protected prior to a transition from active archive to inactive archive, with SHA-1 hashing. Only inactive archive files are hashed and these hashes are recorded by the LogRhythm system for use in verifying integrity during archive restoration. The collected logs are formatted as ASCII text strings and can be encrypted before forwarding across untrusted networks (e.g. Internet). Modification of the archived logs can be detected by rehashing and comparing the values. Note that the SHA-1 hash values are calculated by the operational environment and stored in the EM database. The TOE obtains time from the operational environment and uses this time to apply a timestamp to audit and system log records. The TOE normalizes the time to account for time zone differences.

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

- FPT_ITT.1: The TOE uses FIPS 140-2 SSL to protect data transmitted between the TOE components from unauthorized disclosure and modification.

6.1.5 IDS Component Requirements

The TOE meets the requirements specified in the *U.S. Government Protection Profile Intrusion Detection System System for Basic Robustness Environments,* Version 1.7, July 25, 2007. The IDS Components requirements that are specified in the PP and that the TOE conforms to are summarized as follows. The TOE monitors an IT System for activity that may inappropriately affect the IT System's assets. Information is gathered by the TOE System Monitor Agents and sent to a Mediator Server which analyzes the data against defined rules. A Mediator Server may send log metadata to an AI Engine Server for additional analysis of sets of log messages over time. The analysis process includes identifying and categorizing the log messages and determining if they will be forwarded to the EM as an Event. Once identified as an Event, the TOE determines a course of action such as sending an alarm, sending SNMP traps, report generation, etc. Details of how the TOE meets the PP requirements follow.

Sensor/Scanner:

The TOE is capable of collecting the following events:

- Start-up and shutdown;
- identification and authentication events;
- data accesses;
- service requests;
- network traffic;
- security configuration changes;
- data introduction;
- detected malicious code;
- access control configuration;
- service configuration;
- authentication configuration;
- accountability policy configuration; and
- detected known vulnerabilities. (IDS_SDC.1)

The System Monitor Agents collect logs from all log sources and forwards them in real-time to the Log Manager where they are stored. SQL Servers store audit data in trace files, which are in a Microsoft proprietary binary format. The Trace File Converter service of the TOE converts such trace files from binary to text format, which a System Monitor Agent can then process. Each collected occurrence is called a "log". A Mediator Server processes each log and identifies log metadata, which may be forwarded to an AI Engine Server. Each collected log is also archived to a separate database on the Log Manager where they are hashed using SHA-1. (IDS_STG.1(1)). The Archive file hashes are recorded by the LogRhythm system for use in verifying integrity. Modification of the archived logs can be detected by rehashing and comparing the values. Note that the SHA-1 hash values are stored separately in the EM database. The logs are protected from modification by not providing any interfaces to modify them. In addition the log files are protected from unauthorized deletion by restricting the interfaces that allow the logs to be purged. These interfaces are restricted to users with the authorized Global Admin administrative role.

Each log source is associated with an archive policy. This policy dictates whether logs received from a particular source should be archived or not. The log is analyzed and if the policy dictates the log should be archived it is written to the active archive file. During analysis, the Mediator Server compares the log against the defined knowledge-base rules to determine if the log should be forwarded to the Event Manager as an 'Event', where it is stored and actions can be taken as a result. An AI Engine Server may compare sets of log metadata against AI rules to identify complex Events. Each log and event file entry contains the date and time of the event (or log), subject identity, and the outcome of the event. In addition, some logs contain location, service, protocol information; source and destination addresses; and other information specific to the type of log collected (see system event table for a complete list of information the TOE captures for each type of log). The log, archive and event databases can all be viewed and searched by authorized administrators. The Console provides authorized administrators with the ability to view and search this data via the LogRhythm Investigator, a tool which displays results in 3-D graphical representation. The LogRhythm Investigator can be used for searching and viewing specific sets of logs and events, such as those associated with a specific user, set of users, specific IP address or range, impacted hosts, impacted applications, date and time, and more. Once defined, investigation criteria can be saved and used again. Investigations can include Events, log metadata, raw log data or any combination thereof.

Analyzer:

The TOE performs signature and integrity analysis on all log files received. The analytical results are stored in the EM database. Each record contains various information including date and time of the result (same as date and time of the log), identification of data source and event specific data (IDS_ANL.1). The LogRhythm ARM service, which resides on the EM, is responsible for processing alarm rules against incoming Events and taking the

appropriate action. The TOE can be configured to send SNMP traps, SMTP emails, and perform remediation actions. A remediation action is an external executable or script invoked by the ARM service. Either an Alarm Rule or an AI Engine Rule triggers a remediation action. An action can be configured to take place immediately or to defer execution until approved. An authorized System administrator can configure the timing and approvers for each remediation action. LogRhythm provides remediation action as plug-ins for common actions and supports custom plug-ins (IDS_RCT.1). Authorized administrators can view, and work with, alarm notification policies via the My LogRhythm menu on the Console. The alarm rules define criteria that an Event must satisfy in order to generate an alarm. The visible notification policy for each user is restricted to those policies privately belonging to the currently logged in user. The analyzer and system logs and events can be viewed from the console by users with the Global Admin, Global Analysts, and Restricted Analyst administrative roles. The logs and Events are provided in a readable format to authorized users (IDS_RDR.1). Updates to the Knowledge Base rules can be obtained by licensed customers at the vendor's website. Only authorized Global Admin administrators are permitted to download these updates.

There are two types of resources that affect log collection when exhausted: allocated storage and available storage. A Global Admin allocates storage to components such as database storage for a LM or disk storage for an AI Engine Server. In addition, the operating system provides storage to meet the Global Admin's allocation. The TOE may exhaust either type of storage. For example, LM log storage would be affected when either a LM database reaches the configured limit or LM database size is below the limit but the operating system lacks available disk space to provide to the database.

A potential loss of logs is prevented by the layered architecture of the TOE's solution and by providing administrative interfaces to configure database sizes and automatic purging scripts. For systems with high a volume of logs, more than one LM is configured to accept the incoming logs. If one LM database becomes 90% full, then the LM Mediator Server service is suspended and the System Monitor Agents will send the logs to another LM. The Mediator Server service will go into a 'suspend' mode if the LM database reaches 90%, and stop accepting incoming connections from System Monitor Agents, effectively protecting the Mediator Server service from exhausting the storage in the LM database that it has been allocated. Once suspended, the LogRhythm Mediator Server service writes an event to its internal log stored on a separate volume. The LogRhythm Mediator Server service log displays a WARNING in the file indicating that the system is in a 'suspend' mode. If all LMs are suspended, then the System Monitor Agent writes the logs to disk and forwards them once the LM database has been purged by an authorized administrator (IDS_STG.1, IDS_STG.2). Logs also get written to disk if the System Monitor Agent is not connected to the LM and are forwarded once the connection is re-established. The Global Admin can configure the system to issue a warning after a specified interval of time has elapsed with no log collection.

Purging consists of clearing the logs from the LM database. Note that it is rare that this happens because the TOE is designed to handle large volumes of log data and because the tools are provided to the administrator to configure the pruning and aging scripts, which are designed to automatically manage the database.

The TSF is a distributed system. Each System Monitor Agent has local, persistent storage for log data, which is used when the System Monitor Agent cannot contact an assigned Log Manager. When local storage is exhausted, as defined by the minimum threshold of 1GB, the System Monitor Agent will stop collecting. Similarly, when available local storage falls below the minimum threshold of 1 GB, a Mediator will suspend processing of all incoming logs. The Agent will log an error indicating that data will be thrown away until more disk space becomes available. The error message serves as an alarm that the agent is no longer collecting logs. The Mediator too will log an error indicating a suspend condition resulting from minimum disk space. Once disk space is completely exhausted on an LM, SQL Server will shut down and all LogRhythm components will cease to operate.

All remote Agents will spool incoming logs to disk until such time as they can reconnect to the Mediator.

To prevent local storage exhaustion on a Mediator, an administrator should create a Windows Performance Counter to monitor and alert on free disk space. LogRhythm can then be configured to collect the corresponding Event Log and send alarm notification when the disk reaches the defined minimum threshold. This threshold must exceed the 1GB minimum at which logs will no longer be collected.

In the event all storage space (database and underlying disk sub-system) is exhausted on each LM that a System Monitor Agent is configured to send logs to, the agent behavior varies based on the log collection interface:

1. For interface types that are read by the System Monitor Agent including flat ASCII files, Windows Event Logs, database logs (residing in a database table), and Cisco SDEE sources, the System Monitor Agent will suspend collection relying on the logging systems to buffer logs until the System Monitor Agent can resume collection as individual LMs come back online.
2. For interface types that push data to the System Monitor Agent including syslog (UDP and TCP), Cisco NetFlow, and Checkpoint LEA interface, the System Monitor Agent will continue to accept log data and write to local storage until the local storage is exhausted. Upon the exhaustion of local storage, the System Monitor Agent will typically cease to function. Logs that are pushed to the System Monitor Agent are at greatest risk of loss (e.g. syslog which only exists on the network until reception at a System Monitor Agent). When LMs come back online the System Monitor Agent will continue sending current data and will periodically read in the data written to disk and forward to an LM.

An AI Engine Server handles System Data storage exhaustion by deleting oldest System data. Each AI Engine Server has local, persistent storage where it buffers log data files it receives from Log Managers. The AI Engine reads the log data files, processes them, and then deletes the data files from the file system. If the size of the data files exceeds a configurable amount, the AI Engine Server begins to delete the oldest data files while continuing to write the newest logs to new data files on the file system. The AI Engine Server writes logs to the Windows Event Log indicating that the deletions are occurring, which serves as an alarm that when System data storage is exhausted (IDS_STG.1(2) and IDS_STG.2(2)).

Event information is delivered in real time to Console personal dashboards of those users predefined as authorized viewers for those classifications of Events. Through the personal dashboard, users can monitor events in real time. In addition, the analyzed log results identified as Events (less than 1% of raw log data) are stored in a database on the EM. These log files can be re-created by choosing the raw log data from the LM on which the data resides and re-applying the Knowledge-Base rules. In addition, the same log files can be run against newly updated Knowledge-Base rules. The size of the EM database should be configured according to the product administrative guides in order to provide sufficient storage space for expected logs. The database is designed to automatically expand storage space when log space is needed. All unallocated storage is used by the operating system, database, and LogRhythm processes. Storage expansion options are available. All appliances can be expanded via LogRhythm Direct Attached Storage (DAS), LogRhythm Network Attached Storage (NAS), or the LRSA2 and LRSA4 storage appliances.

The IDS Component function is designed to satisfy the following security functional requirements:

- IDS_ SDC_EXT.1: The TOE is able to collect system data from the targeted IT System resources and records various details of the event.
- IDS_ANL_EXT.1: The TOE performs signature and integrity analysis on all IDS data received and records various information about the analytical result.
- IDS_RCT_EXT.1: The TOE can send alarms to the alarm database and also to users configured to receive alarms.
- IDS_RDR_EXT.1: The TOE provides authorized administrators with the ability to read analyzer data and restricts this capability to these authorized administrators.
- IDS_STG_EXT.1(1): The TOE protects the stored data from unauthorized deletion and modification. System Monitor Agents, Log Managers, and the Event Manager It preserve existing System data in the event of storage exhaustion.
- IDS_STG_EXT.1(2): The TOE protects the stored data from unauthorized deletion and modification. AI Engine Servers preserve newest System data in the event of storage exhaustion.
- IDS_STG_EXT.2(1): System Monitor Agents, Log Managers and the Event Manager The TOE components ignore system data and send alarms if storage capacity is reached.
- IDS_STG_EXT.2(2): AI Engine Servers overwrite oldest stored System data and send alarms if storage capacity is reached.

7. Protection Profile Claims

The TOE conforms to the *U.S. Government Protection Profile Intrusion Detection System for Basic Robustness Environments,* Version 1.7, July 25, 2007.

The TOE provides administrators with the capability to sort the audit records based on date and time, subject identity, type of event, and success or failure of related event. Consequently, the ST changes OE.AUDIT_SORT from an objective for the operational environment to O.AUDIT_SORT an objective for the TOE. The content of the objective is unchanged and is still met by TOE requirement FAU_SAR.3. The change in security objectives is allowed since modified statement of security objectives is more restrictive that the statement of security objectives in the IDSSPP.

This Security Target includes all of the Security Functional and Security Assurance Requirements from the PP verbatim, except as noted below:

> The ST Author added two new security functional requirements (as required by CCEVS policy #13 and precedence). FIA_UID.1 and FIA_UAU.1 were strengthened to .2. Four requirements identified in the PP were omitted from the ST because they do not pertain to this TOE ("**Omitted**"). The other requirements are copied word-for-word from the PP (except where the requirement wording was modified in CC V3.1 R2) with the appropriate selections and assignments made as necessary. The table below identifies which requirements were added ("**Addition**"), and which requirements were copied ("PP Verbatim or CC V3.1 R2 Verbatim"). The rationale for any addition or modification is provided in column 3.

Requirement Component	PP Conformance	Addition/Modification Rationale
FAU_GEN.1: Audit Data Generation	CC V3.1 R2 Verbatim	Changed the PP wording to be compliant with CC v3.1, Revision 2. In addition, auditable events associated with SFRs added to the ST but not specified in the PP have been included in Table 2.
FAU_SAR.1: Audit review	PP Verbatim	Completed the required assignment operation.
FAU_SAR.2: Restricted audit review	PP Verbatim	Not applicable
FAU_SAR.3: Selectable audit review	CC V3.1 R2 Verbatim	Changed the PP wording to be compliant with CC v3.1, Revision 2.
FAU_SEL.1: Selective Audit	CC V3.1 R2 Verbatim	Changed the PP wording to be compliant with CC v3.1, Revision 2.
FAU_STG.2: Guarantee of audit data availability	CC V3.1 R2 refined	Completed the required assignment and selection operations. The assignment "detect" was refined to "prevent" which is more restrictive and thus allowed. Changed the PP wording to be compliant with CC v3.1, Revision 2.

Requirement Component	PP Conformance	Addition/Modification Rationale
FAU_STG.4: Prevention of audit data loss	PP Verbatim	Selection was made and refined, which makes the requirement more restrictive
FIA_AFL.1: Authentication Failure Handling	**Omitted**	**per PD-0097:** LogRhythm does not provide a capability for external IT products to connect to it, therefore this requirement is not applicable.
FIA_UAU.1: Timing of authentication	Replaced with FIA_UAU.2 - CC V3.1 Verbatim	FIA_UAU.2 augments the PP requirement FIA_UAU.1 strengthening the requirement. FIA_UAU.2 is hierarchically stricter than the required FIA_UAU.1.
FIA_ATD.1: User attribute definition	PP Verbatim	Not applicable
FIA_UID.1: User Identification Before Any Action	Replaced with FIA_UID.2 - CC V3.1 Verbatim	The TSF does not allow any access to the TOE prior to Identification and Authentication and therefore FIA_UID.2 was chosen. FIA_UID.2 is hierarchically stricter than the required FIA_UID.1.
FMT_MOF.1: Management of security functions	PP Verbatim	Not applicable
FMT_MTD.1: Management of TSF data	PP Verbatim	Completed the required assignment operation.
FMT_SMF.1: Specification of management functions	**Addition**	Dependency for FMT_MOF.1 as specified in the CC V3.1 R2
FMT_SMR.1: Security roles	PP Verbatim	Completed the optional assignment operation.
FPT_ITA.1: Inter-TSF availability within a defined availability metric	**Omitted per PD-0097 – N/A**	**Requirement is not applicable per PD-0097** The TOE does not communicate with IDS System components outside the TOE, and therefore these requirements have been removed from the PP.
FPT_ITC.1: Inter-TSF confidentiality during transmission	**Omitted per PD-0097 – N/A**	**Requirement is not applicable per PD.** The TOE does not communicate with IDS System components outside the TOE, and therefore these requirements have been removed from the PP.

Requirement Component	PP Conformance	Addition/Modification Rationale
FPT_ITI.1: Inter-TSF detection of modification	**Omitted per PD-0097 – N/A**	**Requirement is not applicable per PD** The TOE does not communicate with IDS System components outside the TOE, and therefore these requirements have been removed from the PP.
FPT_ITT.1: Basic internal TSF data transfer protection	**Addition** – Replaces FPT_ITA.1, FPT_ITC.1, and FPT_ITI.1 from the PP per PD-0097	**Requirement added per PD-0097**
FPT_STM.1: Reliable time stamps	**Omitted per PD-0151**	The precedent states that in the PP; "OE.TIME The IT Environment will provide reliable timestamps to the TOE" and that FPT_STM.1: Reliable time stamps should have been designated to the IT environment. Therefore, it is acceptable to get the timestamps from the TOE and/or from an external NTP server as long as the appropriate requirements are included in the ST. In this TOE the timestamp is obtained from the operational environment.
IDS_ANL.1: Analyzer analysis	PP Verbatim	Required Selection made
IDS_RCT.1: Analyzer react	PP Verbatim	Required Assignments made
IDS_RDR.1: Restricted data review	PP Verbatim	Required Assignments made
IDS_STG.1: Guarantee of analyzer data availability	PP Verbatim	Iterated to specify behaviors of distributed TOE components. Required Assignment and selection made
IDS_STG.2: Prevention of Analyzer data loss	PP Verbatim	Iterated to specify behaviors of distributed TOE components. Required selection made
IDS_SDC.1: System Data Collection	PP Verbatim	Required selection and assignment made

Table 5 - Protection Profile Claims

The IDSS Protection Profile specifies EAL2 augmented with, ALC_FLR.2 (Flaw Remediation).

The level of assurance chosen for this ST is that of Evaluation Assurance Level (EAL) 2, as defined in [CC] Part 3, augmented with the [CC] Part 3 component ALC_FLR.2. The assurance requirements in this ST are therefore equal to the ones required by the claimed PPs.

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;
- Requirement Dependencies;
- TOE Summary Specification; and
- PP Claims

8.1 Security Objectives Rationale

The *U.S. Government Protection Profile Intrusion Detection System For Basic Robustness Environments* (IDSSPP) provides rationale for the security objectives demonstrating that security objectives are suitable to cover the intended environment. The reallocation of OE.AUDIT_SORT (for the operational environment) to O.AUDIT_SORT (for the TOE) does not change the content of the security objective. The rationale (provided in Sections 6.1 and 6.2 of the IDSSPP) is valid for the PP objectives reproduced in this ST and is not further discussed.

8.2 Security Functional Requirements Rationale

Section 6.3 of the IDSSPP provides rationale for the security functional requirements, demonstrating that the security functional requirements are suitable to address the TOE security objectives. This rationale is valid for the PP requirements reproduced in the ST and is not further discussed.

This ST includes the following security functional requirements not included in the IDSSPP: FMT_SMF.1; and FPT_ITT.1. The following table maps these requirements to applicable TOE security objectives described in Section 4. Supporting rationale for these mappings is provided following the table.

	O.EADMIN	O.INTEGR
FMT_SMF.1	**X**	
FPT_ITT.1		**X**

Table 6 - Requirement to Objective

8.2.1.1 O.EADMIN

The TOE must include a set of functions that allow effective management of its functions and data.

The following security functional requirements contribute to satisfying this security objective:

- FMT_SMF.1—the ST includes FMT_SMF.1 to specify the security management functions that are required to provide the capabilities for effective management of the TOE's functions and data

8.2.1.2 O.INTEGR

The TOE must ensure the integrity of all audit and System data.

The following security functional requirement contributes to satisfying this security objective:

- FPT_ITT.1—the ST includes FPT_ITT.1 to specify that the TOE will protect the all audit and System data during transmission between separate parts of the TOE.

8.3 Security Assurance Requirements Rationale

The IDSSPP provides rationale for the security assurance requirements, demonstrating that they are sufficient given the statement of security environment and security objectives. The rationale is provided in Section 6.4 of the IDSSPP and is valid for this ST as no new security environment statements or objectives were.

EAL2 was selected as the assurance level because the TOE is a commercial product whose users require a low to moderate level of independently assured security. The TOE is targeted at an environment with good physical access security where it is assumed that attackers will have low attack potential. Augmentation was chosen to provide the added assurance that is provided by defining flaw remediation procedures. Therefore, the target assurance level of EAL 2 augmented with ALC_FLR.2 is appropriate for such an environment.

8.4 Requirement Dependency Rationale

The dependency requirements rationale is presented in Section 6.7 of the IDSSPP. The IDSSPP requirements have been evaluated and it has been determined that all dependencies have been satisfactorily addressed in the IDSSPP.

This ST includes the following security functional requirements not included in the IDSSPP: FMT_SMF.1; and FPT_ITT.1. The following table demonstrates how the dependencies of each of these additional SFRs are satisfied in the ST.

ST Requirement	CC Dependencies	ST Dependencies
FMT_SMF.1	None	None
FPT_ITT.1	None	None

Table 7 - Requirement Dependencies

8.5 TOE Summary Specification Rationale

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.

This Section in conjunction with Section 6, the TOE Summary Specification, provides evidence that the security functions are suitable to meet the TOE security requirements. The collection of security functions work together to provide all of the security requirements. The security functions described in the TOE summary specification are all necessary for the required security functionality in the TSF.

8.6 PP Claims Rationale

See Section 7, Protection Profile Claims.