LogRhythm Integrated Solution v7.8 with Microsoft SQL Server 2016 SP1 Standard Edition

Security Target

Version 1.0
May 13, 2022

Prepared for:

LogRhythm Inc.
4780 Pearl East Circle
Boulder, CO 80301

Prepared by:

Leidos, Inc.
Common Criteria Testing Laboratory
6841 Benjamin Franklin Drive, Columbia, Maryland 21046

LIST OF 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 LogRhythm Integrated Solution provided by LogRhythm Inc. The TOE is delivered as a software only solution that provides log and event management, file integrity monitoring, and endpoint monitoring and control. The TOE consists of several components that coordinate with one another to collect and analyze information from multiple log sources including syslog, snmp, netflow and sflow devices, Windows events, flat file, databases or applications.

The Security Target contains the following additional sections:

- TOE Description (Section2)
- Security Problem Definition (Section 3)
- Security Objectives (Section 4)
- IT Security Requirements (Section 5)
- TOE Summary Specification (Section 6)
- Rationale (Section 7)

1.1 Security Target, TOE and CC Identification

ST Title – LogRhythm Integrated Solution v7.8 with Microsoft SQL Server 2016 SP1 Standard Edition Security Target

ST Version – Version 1.0

ST Date – May 13, 2022

TOE Identification – LogRhythm Integrated Solution v7.8.0 with Microsoft SQL Server 2016 SP1 Standard Edition

A deployment of the LogRhythm Integrated Solution consists of:

LogRhythm Integrated Solution software consisting of:

- 1 or more Data Processor(s) Software
- 1 Platform Manager Software
- 1 or more Data Indexer(s) Software
- 1 or more System Monitors Software
- 0 or more Data Collectors Software
- 1 or more Advanced Intelligence (AI) Engines Software
- 1 or more instances of LogRhythm Client Console Software
- 1 or more instances of LogRhythm Web Console Software
- 1 Microsoft SQL Server 2016 SP1 Standard Edition

TOE Developer – LogRhythm Inc.

Evaluation Sponsor – LogRhythm, Inc.

CC Identification – Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 5, April 2017

1.2 Conformance Claims

This TOE is conformant to the following CC specifications:

- Common Criteria for Information Technology Security Evaluation Part 2: Security functional components, Version 3.1, Revision 5, April 2017.
    - Part 2 Extended
- Common Criteria for Information Technology Security Evaluation Part 3: Security assurance components, Version 3.1 Revision 5, April 2017.
    - Part 3 Conformant
    - Assurance Level: EAL 2 Augmented (ALC_FLR.2).

1.3 Protection Profile Conformance Claim

The TOE does not claim conformance to a Protection Profile.

1.4 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 number in parentheses 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 (for example, [**assignment**]). Note that an assignment within a selection would be identified in italics and with embedded bold brackets (for example, [***[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 (for example, [***selection***]).
    - Refinement: allows the addition of details. Refinements are indicated using bold, for additions, and strike-through, for deletions (for example, "… **all** objects …" or "… ~~some~~ **big** things …"). Note that 'cases' that are not applicable in a given SFR have simply been removed without any explicit identification.
- Other sections of the ST – Other sections of the ST use bolding to highlight text of special interest, such as captions.

1.5 Glossary

Acronym	Description
AIE	LogRhythm Advanced Intelligence Engine
CMDB	Case Management Database
DP	LogRhythm Data Processor
DX	LogRhythm Data Indexer
EAL	Evaluation Assurance Level
FTP	File Transfer Protocol

Acronym	Description
LDAP	Lightweight Directory Access Protocol
MPS	Messages per Second
MPD	Messages per Day
MPE	Message Processing Engine
MS-TDS	Microsoft Tabular Data Stream Protocol
NTP	Network Time Protocol
PM	LogRhythm Platform Manager
SA	System Administrator
SDEE	Security Device Event Exchange
SIEM	Security information and event management
SFR	Security Functional Requirement
TLS	Transport Layer Security
TOE	Target of Evaluation
TSF	TOE Security Functionality
UDLA	Universal Database Log Adapter

Table 1 - Acronyms

1.6 Terminology

The terminology below is described in order to clarify the terms used in the ST as well as those used in the TOE product documentation.

Term	Definition
Alarm Rules	A tab within Deployment Manager used to manage Alarm Rules. An Alarm Rule is a configuration of information to identify events that should trigger an alarm to be created. It is similar to the analysis tools in that it provides for the user to specify primary and log source criteria, field filters, and day and time criteria, along with other alarm specific settings such as aggregation, suppression, notifications, and remediation actions.
Classification	A second tier group used to categorize logs and events. There is one or more Classification associated to a Classification Type. A Classification can have one or more Common Events, which are associated to an MPE Rule.
Classification Type	A first tier group used to categorize logs and events. There are three types: Audit, Security, and Operations. Classifications are grouped into one of these three types. There is one classification type for one or more classifications.
Common Event	A short, plain-language description of the log that is associated to a specific classification. There is one classification for one or more Common Events. A Common Event is created and managed through the Common Event Manager, and Common Events are associated with MPE rules (base and sub) in the MPE Rule Builder. There is a one-to-one relationship between an MPE rule and a Common Event.
Custom Objects	A type of Object that is created by an end user.
Deployment Manager	A utility window in the LogRhythm Console. Users with LogRhythm administrator credentials use it to configure and manage LogRhythm components and functionality such as alarming and reporting.

Term	Definition
ElasticSearch	ElasticSearch is an open source, RESTful search engine built on top of Apache Lucene and released under an Apache license. It is Java-based and can search and index document files in diverse formats.
Entities	An entity represents a physical location in a deployment, such as network records, and host records, and LogRhythm components. The Entities tab opens by default when you access the Deployment Manager from the Console.
Event	A log having more immediate operational, security, or compliance relevance. Typically logs classified as errors, failures, or attacks are considered events.
Knowledge Base	A LogRhythm package that includes both required and optional content shared across a LogRhythm deployment. It consists of the core Knowledge Base as well as modules. The core Knowledge Base includes content applicable to all deployments, such as log processing rules, policies, and classifications.
SEM Data	Refers both to raw data collected by the TOE and to the results of analysis applied by the TOE to that data.
Log Sources	Log Sources are single, unique origins of log data that is collected from a Host and is assigned a Message Processing Engine (MPE) policy. A single Host can have multiple Log Sources. A Log Source is the key link LogRhythm uses to determine a log message's origin.
LogRhythm ARM	LogRhythm Alarming and Response Manager (ARM) Service. The ARM processes alarm rules and takes the appropriate response, such as sending email to people on a notification list.
LogRhythm LogMart	The Platform Manager Database that tracks unique log messages with aggregated occurrence information for collected log data. Includes tables that track the volume of log data collected from each log source.
LogRhythm EMDB	The Platform Manager Database that stores the following information: all configuration information for a LogRhythm deployment (entities, hosts, networks, agents, log sources, and so on); all records of the archive file properties such as physical properties and hash for verification; all Knowledge Base information that is imported via the KB import process (MPE Rules/Policies, Investigations, Reports, etc.).
LogRhythm Alarms	The Platform Manager Database that stores all alarms, alarm notifications, and alarm histories generated by the LogRhythm ARM.
LogRhythm CMDB	The Platform Manager Database that stores the Case Management data from the LogRhythm Web Console.
LogRhythm Events	The Platform Manager Database that stores all events - log data and metadata - that have been forwarded from Data Processors.
Metadata	Metadata fields store network and host information pulled from the log message.
Microsoft Tabular Data Stream Protocol	The Tabular Data Stream (TDS) protocol is an application layer request/response protocol that facilitates interaction with a database server and provides for the following: • Authentication and channel encryption negotiation. • Specification of requests in SQL (including Bulk Insert). TDS depends on Transport Layer Security (TLS)/Secure Socket Layer (SSL) for network channel encryption. The TDS protocol depends on TLS/SSL to encrypt data transmission.
Object	Resource such as a file, file path, or registry key that is referenced or impacted by log activity.

Term	Definition
System Monitor	System Monitors collect and forward log data to Data Processors.
System Objects	A type of object that is created by LogRhythm Labs and imported with the Knowledge Base.
Tail	A monitoring analysis tool that provides real-time monitoring of log and event activity. It is an easy means of monitoring any activity based on device, log classification, or metadata contained in the log.
Private	A permission value for an object that can only be managed by the owner.
Public	A permission value for an object that generally means all users have access to view the object. May have edit restrictions.
Global	A permission value that provides general global access for the object.
Windows Performance Counter	An application within a Microsoft Windows operating system or Windows applications / services that provides parameters which can be monitored in real-time.

Table 2 - Terminology

2. TOE Description

The Target of Evaluation (TOE) is the LogRhythm Integrated Solution v7.8 with Microsoft SQL Server 2016 SP1 Standard Edition software. The TOE is a fully integrated Security Information and Event Management (SIEM) solution that 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 the LogRhythm Integrated Solution consists of:

- 1 or more Data Processor(s) Software
- 1 Platform Manager Software
- 1 or more Data Indexer(s) Software
- 1 or more System Monitors Software
- 0 or more Data Collectors Software
- 1 or more Advanced Intelligence (AI) Engines Software
- 1 or more instances of LogRhythm Client Console Software
- 1 or more instances of LogRhythm Web Console Software
- 1 Microsoft SQL Server 2016 SP1 Standard Edition

The TOE consists of several software components that coordinate with one another to provide automated centralization of log collection and event management. The TOE collects information from multiple log sources (such as Windows events, syslog, flat file, NetFlow, sFlow, databases or applications).

The TOE is evaluated as a software only application. However, for convenience the TOE can be delivered on pre-configured dedicated LogRhythm appliances. The appliance hardware (if purchased in this configuration) is not part of the TOE.

2.2 TOE Architecture

The following figure depicts the TOE components within their environment and shows communications among the components. Microsoft SQL Server 2016 SP1 Standard Edition is an internal component of the Platform Manager, and is not shown in the figure.

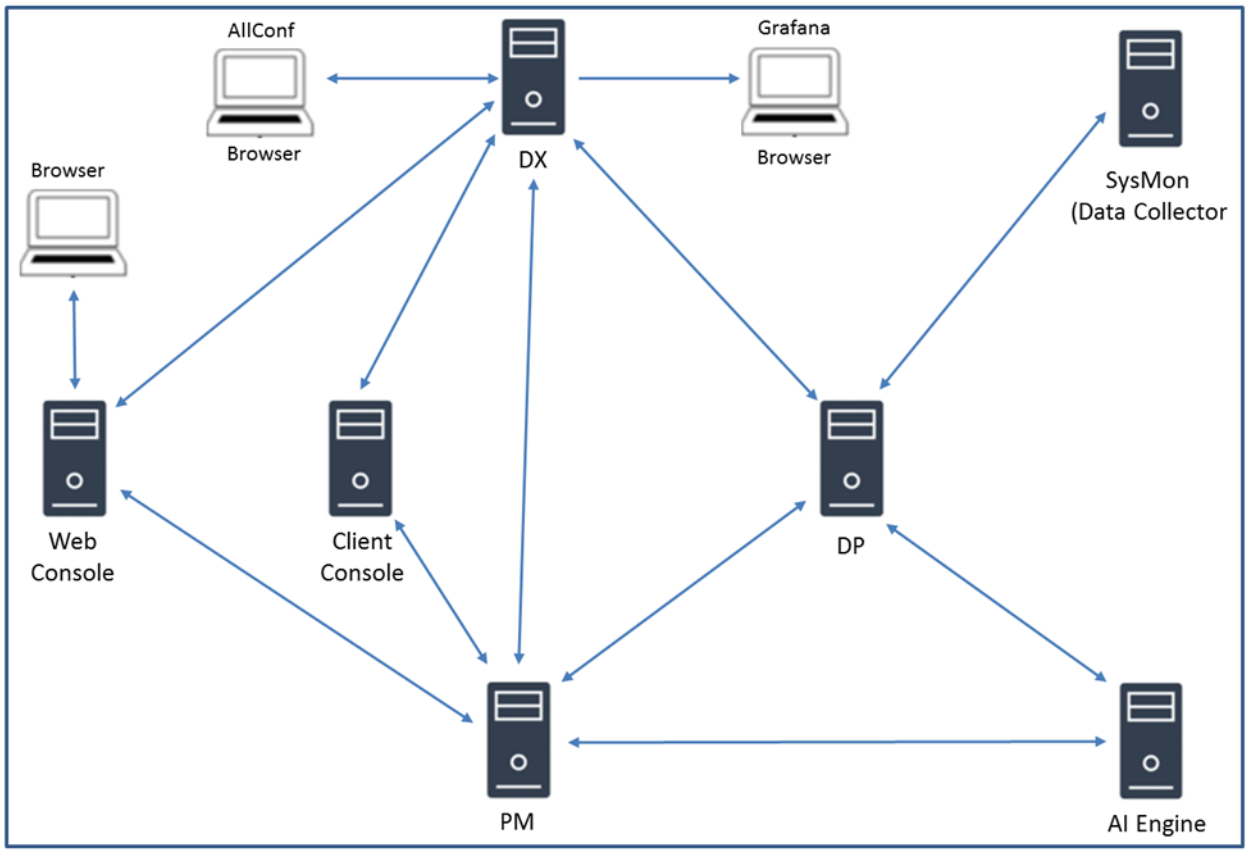

Figure 1 – TOE Deployment

The LogRhythm System Monitor(s), Data Processors(s), AI Engine Server(s), Platform Manager, Client Console(s), Web Console, Data Indexer, Data Collector, and Microsoft SQL Server 2016 SP1 Standard Edition software constitute the TOE. The TOE can be purchased as software only or for convenience the TOE can be delivered on pre-configured dedicated LogRhythm appliances. The appliance hardware (if purchased in this configuration) is not part of the TOE. These options are described in detail in Section 2.2.1 below.
The modular nature of the LogRhythm components allows the Platform Manager, Data Processor, and Data Indexer to reside on the same server for low-volume deployments, or on dedicated servers for high volume deployments.

The System Monitors can:

- Be deployed on supported Windows, Linux, Solaris, AIX, or HP-UX systems in an internal DMZ, or Remote Network machines.
- Encrypt collected log data before forwarding it across untrusted networks (for example, internet).
- Read collected log data before forwarding it across untrusted networks (for example, flat files).
- Read local Windows Event Logs (Windows version).

- Read Windows Event Logs residing on remote systems (Windows version).
- Have an integrated Syslog server for collection of Syslog data.
- Have an integrated NetFlow server (Windows version) for collecting NetFlow/J-Flow data from Cisco devices.
- Have an integrated sFlow server for collecting Flow data from network devices.
- Have an integrated SNMP trap receiver.
- Have additional capabilities built in to collect
    - Box events
    - eStreamer logs
    - SalesForce logs
    - Office365 logs
    - IP360 logs
    - Okt logs
    - CloudTrail logs
    - CloudTrailS3 logs
    - CloudWatch logs
    - NetCloud logs
    - Tenable.io logs
    - Tenable Security Center logs
    - S3Access logs
    - Check Point Firewall logs
    - Cisco IDS logs
    - Remote Database UDLA Logs
    - Nexpose, Metasploit, Retina, Qualys and Nessus vulnerability logs

The optional Data Collector Appliance provides remote, high-performance collection of all machine data, including log messages, application data, security events, and network flows. They encrypt, compress and transport data from remote locations to LogRhythm Data Processors, either in real time or on a schedule.

In general, remote log information flows to the System Monitor/Data Collector through the Data Processor to the AI Engine Server to the Platform Manager where a SQL Server is used internally to store log and event information. The Data Processor forwards logs to the Data Indexer where Elastic Search is used internally to store log information. TLS is used when receiving logs at the Data Processor from the LogRhythm System Monitors and also when sending logs from the Data Processor to AI Engine and Data Indexer. System Monitors collect log messages. Data Processors 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. The Data Processor processes Events and raises alarms as appropriate. AI Engine Servers analyze log metadata gleaned from sets of log messages to identify more complex Events. The Data Indexers provide highly scalable indexing and searching of machine and forensic data. Indexers store both the original and structured copies of data to enable search-based analytics.

The System Monitor is a software component that provides local and remote log data collection across various operating systems including Windows, Linux, AIX, HPUX, and Solaris. The System Monitor is a central log data collector, collecting logs from many devices, servers, databases, and applications, performing host activity monitoring and forwarding logs to the Data Processor. The System Monitor converts collected logs to ASCII text strings, which can be encrypted before forwarding across untrusted networks (e.g. Internet). SQL Trace File

Converter is a support service, which converts binary SQL Server trace files into UTF8 encoded text files that LogRhythm Windows and UNIX System Monitors can read and forward to the Data Processor for processing. Each System Monitor forwards logs to the Data Processor that is configured to receive them, where they are analyzed against defined Knowledge Base rules. System Monitor communications with Data Processor(s) are authenticated and encrypted via TLS.

Each Data Processor is a Windows server that consists of a LogRhythm Mediator Server service. There can be one or more Data Processors per deployment to provide event processing and forwarding. The Data Processor Mediator Server contains the Message Processing Engine (MPE) and the AI Engine Data Provider. The MPE processes logs against rules in the 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 Platform Manager as an Event or both. The Data Processor's Mediator Server service handles communications with LogRhythm System Monitors, such as authenticating connections, receiving log data, and informing System Monitors to shut down or failover when required. The Mediator Server is also responsible for sending raw and processed log messages to the Data Indexer.

The Data Indexer is a Windows or Linux server, and it should be protected with strict access controls placed on devices that can connect to the log repository if deployed in a DMZ or an untrusted environment. Communications to DMZ or remotely deployed Data Processors, from Platform Managers and Consoles, can be encrypted to provide secure log delivery. The Data Indexer provides high-performance, distributed, and highly scalable indexing and searching of machine and forensic data based upon ElasticSearch. Data Indexers store both the original and structured copies of data to enable search-based analytics. The Data Indexer is configured via a browser connecting to the local Data Indexer AllConf service for initial configuration. Once the Data Indexer is placed into the evaluated configuration; the AllConf service is no longer used.

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 Data Processors. It marshals the data for the AI Engine to process. Also, it maintains TLS connections with Data Processors. 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 Platform Manager. There are no databases for the AI Engine. The AI Engine Communication Manager and AI Engine Server run on the AI Engine The AI Engine reads the log data files, processes them, and then deletes the data files from the file system based on the configurable parameter MaxLogDataSize, which is the maximum amount of log data to keep on disk. Each AI Engine Communication Manager has local, persistent storage for the log data files it receives from Mediators.

The LogRhythm Platform Manager is a Windows server running Microsoft SQL Server 2016 SP1 Standard Edition. There is one Platform Manager per deployment to provide centralized event management, incident management, analysis, reporting, and configuration. The Platform Manager houses the Knowledge Base and the following databases: EMDB, CMDB, Alarms, Events, and LogMart. The LogRhythm Alarming and Response Manager (ARM) and LogRhythm Job Manager services run on the Platform Manager.

The Platform Manager provides the central event management and administration of the LogRhythm SIEM, including:

- Configuration information for all System Monitors, log sources, and log source types.
- Knowledge Base, which includes all processing rules, built-in reports (for compliance), built-in alarms, and other processing-related information.
- The Alarming and Response Manager, which is a Windows service responsible for processing alarm rules and taking appropriate response such as sending e-mails to those on a notification list or sending SNMP traps to an SNMP server.
- The Job Manager, which is responsible for scheduled report job generation, System Monitor and Data Processor heartbeat monitoring, Active Directory synchronization, and health monitoring.

The LogRhythm Client Console provides deployment administration and user interaction with a LogRhythm deployment. The Console also provides real-time monitoring, incident management, and interfaces for TOE configuration and user management. Administrators use the Client Console to configure LogRhythm (for example, selecting rules that identify Events) and to view log reports and analyses. The alarms may be viewed by the Web

Console or Client Console. Optionally the alarms may be sent to an external SMTP Server or SNMP Server in the operational environment. The Client Console is a Windows .NET-based client application that can be installed on various Windows operating systems. An AI Engine Server obtains its configuration from the Client Console indirectly via the Platform Manager.

The LogRhythm Web Console allows users to monitor network log activity from supported browsers on desktop computers, laptops, and touch-based tablets. The Web Console provides a customizable user interface with analytical and forensic features.

2.2.1 Physical Boundaries

2.2.1.1 Included Product Components

The TOE consists of the following software components:

- 1 or more Data Processor(s) Software
- 1 Platform Manager Software
- 1 or more Data Indexer(s) Software
- 1 or more System Monitors Software
- 0 or more Data Collectors Software
- 1 or more Advanced Intelligence (AI) Engines Software
- 1 or more instances of LogRhythm Client Console Software
- 1 or more instances of LogRhythm Web Console Software
- 1 Microsoft SQL Server 2016 SP1 Standard Edition

A deployment of the LogRhythm Integrated Solution software components, with the exception of the system monitor agent/data collector, is required to be the same version 7.8.

The Platform Manager, AI Engine Server, Data Processor, Data Collector, Data Indexer 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. The appliance hardware (if purchased in this configuration) is not part of the TOE. 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.

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.

Platform Manager (PM) - The LogRhythm Platform Manager is a Windows server running Microsoft SQL Server 2016 SP1 Standard Edition. There is one Platform Manager per deployment to provide centralized event management, incident management, analysis, reporting, and configuration. The Platform Manager can be installed on a dedicated appliance (recommended for large environments) or on the same system as the DP/DX/AIE for smaller deployments.

The Platform Manager software v7.8 can be delivered pre-installed on a LogRhythm provided appliance with the hardware requirements identified below.

Appliance Series	Hardware Specifications	Operating System
LR-PM5400	CPU - 12 Core Memory - 128 GB RAM Useable/Raw Storage – 2.44 TB / 4.89 TB	Windows 2012 R2 x64 Standard Edition

Appliance Series	Hardware Specifications	Operating System
	Ethernet - 4 x 1 Gigabit Ethernet NICs	
LR-PM5500	1 x 2.2 GHz 10 Core CPU 20 vCPU 128 GB RAM PERC H740 Integrated RAID Controller with 8GB Cache l 2 x 10 Gb/s NICs l 2 x 1 Gb/s NICs	Windows 2016 x64 Standard Edition
LR-PM7400	2 x 2.6 GHz 8 Core CPU (41.6 GHz total) 32 vCPU Dedicated Disk Drives (DAS or SAN) 128 GB RAM PERC H730P Integrated RAID Controller 4 x 1 Gigabit Ethernet NICs	Windows 2012 R2 x64 Standard Edition
LR-PM7500	l 2 x 2.6 GHz 12 Core CPU l 48 vCPU l 196 GB RAM l PERC H740 Integrated RAID Controller with 8GB Cache l 2 x 10 Gb/s NICs l 2 x 1 Gb/s NICs	Windows 2016 x64 Standard Edition

Table 3 – LogRhythm Platform Manager (PM) Appliances

The LogRhythm Platform Manager software v7.8 can be installed on a customer provided platform that meets the above hardware requirements and has the following software installed.

- Windows Server 2012 R2 x64 Standard or Enterprise Edition, or
- Windows Server 2016
- Microsoft .NET Framework 4.5.2 or later
- Microsoft SQL Server 2016 SP1 Standard Edition

Data Processor (DP) - The LogRhythm Data Processor is a Windows Server system. There can be one or more Data Processors per deployment to provide event processing and forwarding. In medium to large installations, Data Processors should be dedicated systems. In low volume deployments, a Data Processor can coexist on the same system as the PM/DX/AIE.

The Data Processor (DP) software can be delivered pre-installed on a LogRhythm provided appliance with the hardware requirements identified below.

Appliance Series	Hardware Specifications	Operating System
LR-DP5400	1 x 2.4 GHz 6 Core CPU (14.4 GHz total)	Windows 2012 R2 x64 Standard Edition

Appliance Series	Hardware Specifications	Operating System
	12 vCPU Dedicated Disk Drives (DAS or SAN) 32 GB RAM PERC H730 Integrated RAID Controller 4 x 1 Gigabit Ethernet NICs	
LR-DP7400	2 x 2.6 GHz 8 Core CPU (41.6 GHz total) 32 vCPU Dedicated Disk Drives (DAS or SAN) 64 GB RAM PERC H730P Integrated RAID Controller 4 x 1 Gigabit Ethernet NICs	Windows 2012 R2 x64 Standard Edition
LR-DP5500	1 x 2.6 GHz 12 Core CPU 24 vCPU 64 GB RAM PERC H740 Integrated RAID Controller with 8GB Cache 2 x 10 Gb/s NICs 2 x 1 Gb/s NICs	Windows 2016 x64 Standard Edition
LR-DP7500	2 x 3.0 GHz 12 Core CPU 48 vCPU 128 GB RAM PERC H740 Integrated RAID Controller with 8GB Cache 2 x 10 Gb/s NICs 2 x 1 Gb/s NICs	Windows 2016 x64 Standard Edition

Table 4 – LogRhythm Data Processor (DP) Appliances

The Data Processor software can also be installed on a customer provided platform that meets the above hardware requirements and has the following software installed.

- Windows Server 2012 R2 x64 Standard or Enterprise Edition, or
- Windows Server 2016
- Microsoft .NET Framework 4.5.2 or later

Data Indexer (DX) - The LogRhythm Data Indexer can be installed on Windows or Linux (CentOS 7.4 minimal) systems. The Data Indexer provides high-performance, distributed, and highly scalable indexing and searching of machine and forensic data. Indexers store both the original and structured copies of data to enable search-based analytics. In medium to large installations, Data Indexers should be dedicated systems. In low volume deployments, a Data Indexer can coexist on the same system as the PM/DP/AIE.

The Data Indexer (DX) software can be delivered pre-installed on a LogRhythm provided appliance with the hardware requirements identified below.

Appliance Series	Hardware Specifications	Operating System
LR-DX3400	1 x 3.0 GHz 4 Core CPU (12 GHz	CentOS 7.4

Appliance Series	Hardware Specifications	Operating System
	total) 4 vCPU Dedicated Disk Drives (DAS or SAN) 32 GB RAM PERC H330 Integrated RAID Controller 2 x 1 Gigabit Ethernet NICs	
LR-DX5401	1 x 2.4 GHz 6 Core CPU (14.4 GHz total) 12 vCPU Dedicated Disk Drives (DAS or SAN) 64 GB RAM PERC H730 Integrated RAID Controller 4 x 1 Gigabit Ethernet NICs	CentOS 7.4
LR-DX7401	2 x 2.4 GHz 6 Core CPU (13.2 GHz total) 24 vCPU Dedicated Disk Drives (DAS or SAN) 128 GB RAM PERC H730P Integrated RAID Controller 4 x 1 Gigabit Ethernet NICs	CentOS 7.4
LR-DX/DN7500	2 x 2.6 GHz 14 Core CPU (72.8 GHz total) 56 vCPU Dedicated Disk Drives (DAS or SAN) 256 GB RAM PERC H740P Integrated RAID Controller 2 x 10 Gigabit Ethernet NICs 2 x 1 Gigabit Ethernet NICs	CentOS 7.4
LR-DX3500 Series	1 x 2.3 GHz 12 Core CPU 24 vCPU 64 GB RAM PERC H740 Integrated RAID Controller with 8GB Cache 2 x 10 Gb/s NICs 2 x 1 Gb/s NICs	CentOS 7.4
LR-DX5500 Series	1 x 2.6 GHz 14 Core CPU 28 vCPU 128 GB RAM	CentOS 7.4

Appliance Series	Hardware Specifications	Operating System
	PERC H740 Integrated RAID Controller with 8GB Cache 2 x 10 Gb/s NICs 2 x 1 Gb/s NICs	

Table 5 – LogRhythm Data Indexer (DX) Appliances

The Data Indexer software can also be installed on a customer provided platform that meets the above hardware requirements and has the following software installed.

- Windows Server 2012 R2 x64 Standard or Windows Server 2016
- Microsoft .NET Framework 4.5.2 or later

  OR
- The Linux Data Indexer can be installed on an existing CentOS 7.4 Minimal system. To simplify the installation, LogRhythm provides an ISO image that contains the CentOS operating system and the Data Indexer installer package.

  New installations of the Data Indexer are only supported on the Linux platform. The Data Indexer is only supported on Windows in an XM configuration.

AI Engine - The AI Engine is a Windows Server system. It is LogRhythm's advanced analysis platform that identifies and categorizes the log messages to determine if they will be forwarded to the Platform Manager as an Event. It provides real-time visibility into risks, threats, and critical operations issues. In medium to large installations, AI Engines should be dedicated systems. In low volume deployments, an AI Engine can coexist on the same system as the PM/DP/DX.

The AI Engine software can be delivered pre-installed on a LogRhythm provided appliance with the hardware requirements identified below.

Appliance Series	Hardware Specifications	Operating System
LR- AIE5400	2 x 2.6 GHz 8 Core CPU (41.6 GHz total) 32 vCPU Dedicated Disk Drives (DAS or SAN) 128 GB RAM PERC H730 Integrated RAID Controller 4 x 1 Gigabit Ethernet NICs	Windows 2012 R2 x64 Standard Edition
LR-AIE7400	x 2.6 GHz 14 Core CPU (72.8 GHz total) 56 vCPU Dedicated Disk Drives (DAS or SAN) 256 GB RAM PERC H730 Integrated RAID Controller 4 x 1 Gigabit Ethernet NICs	Windows 2012 R2 x64 Standard Edition
LR-AIE7500	2 x 3.0 GHz 12 Core CPU 48 vCPU 128 GB RAM PERC H740 Integrated RAID	Windows 2016 x64 Standard Edition

Appliance Series	Hardware Specifications	Operating System
	Controller with 8GB Cache 2 x 10 Gb/s NICs 2 x 1 Gb/s NICs	

Table 6 - AI Engine Appliances

The AI Engine software can also be installed on a customer provided platforms that meets the above hardware requirements and has the following software installed.

- Microsoft Windows Server 2012 R2 x64 Standard or Enterprise Edition, or
- Windows Server 2016
- Microsoft .NET Framework 4.5.2 or later

Web Console - The LogRhythm Web Console allows users to monitor network log activity from supported browsers on desktop computers, laptops, and touch-based tablets. The Web Console provides a customizable user interface with analytical and forensic features.

The Web Console software can be delivered pre-installed on a LogRhythm provided Web Console appliance with the hardware requirements identified below.

Appliance Series	Hardware Specifications	Operating System
LR-WC3400	1 x 2.6 GHz 8 Core CPU 16 vCPU 32 GB RAM H730 RAID controller with 2GB Cache 2 x 1 Gigabit Ethernet NICs	Windows 2012 R2 x64 Standard Edition Or Windows 2016 x64

Table 7 - LogRhythm Dedicated Web Console Appliance

The Web Console can be installed on a server or virtual environment that meets the following specifications:

- Microsoft Windows Server 2012 R2 Standard Edition x64, or
- Windows Server 2016 x64

The Web Console supports the following browsers:

- Google Chrome
- Microsoft Internet Explorer
- Microsoft Edge
- Mozilla Firefox

Client Console - The LogRhythm Client Console provides deployment administration and user interaction with a LogRhythm deployment. Administrators use the Client Console to configure LogRhythm (for example, selecting rules that identify Events) and to view log reports and analyses. The Client Console is a Windows .NET-based client application that can be installed on the following Windows operating systems:

The Client Console can be installed on any system that meets the following minimum requirements.

Hardware

- Available Disk Space: 600 MB
- Memory
    - 1 GB (Administration)

- 2 GB (Analysis & Reporting)

Software

Operating Systems (64-bit)

- Windows 7
- Windows 10 Server
- Windows Server 2008 R2
- Windows Server 2012 R2
- Windows Server 2016
- Microsoft .NET Framework 4.5.2 or later

Data Collector - The optional Data Collector Appliance provides remote, high-performance collection of all machine data, including log messages, application data, security events, and network flows. They encrypt, compress and transport data from remote locations to LogRhythm Data Processors, either in real time or on a schedule. The Data Collector can also be deployed as software. Local, agent-based collection is performed by LogRhythm System Monitors, software that also functions as an endpoint monitor. System Monitors can be installed on servers and virtual machines running Windows, Linux or UNIX. It consolidates and collects log and machine data from remote environments and cloud infrastructure. A single agent functioning as a Data Collector can collect thousands of messages per second from dozens of devices. The System Monitor software can be delivered pre-installed on a LogRhythm provided Data Collector appliance with the hardware requirements identified below.

Appliance Series	Hardware Specifications	Operating System
LR-DC3400 (Data Collector)	1 x 3.0 GHz 4 Core CPU (12 GHz total) 4 vCPU Dedicated Disk Drives (DAS or SAN) 16 GB RAM H330 RAID controller 2 x 1 Gigabit Ethernet NICs	Windows 2012 R2 x64 Standard Edition, or Windows Server 2016

Table 8 – LogRhythm Data Collector Appliances

System Monitors (also called LogRhythm Agents), collect and forward raw log data to the LogRhythm Data Processors. System Monitors can be installed on both Windows and UNIX platforms. They are also integrated into the LogRhythm Data Collector appliances.

The System Monitor software can also be installed on customer provided platforms. The following table lists the operating systems supported by the System Monitor.

The following are the definitions of LogRhythm support levels used in the table.

Certified Support (CS)

- Fully tested per LogRhythm quality assurance processes.
- LogRhythm will patch bugs.
- Full LogRhythm Technical Support.

Limited Support (LS)

- Limited testing, but likely to work based on engineering assessment and/or field verification.
- LogRhythm may patch bugs.
- Limited LogRhythm Technical Support.

LogRhythm System Monitor OS Support Levels		
Operating System	**32-bit/64-bit**	**Support Level**
Windows		
Windows 7	32-bit	LS
Windows 7	64-bit	CS
Windows 8	32-bit, 64-bit	CS
Windows 8.1	32-bit, 64-bit	CS
Windows 10	32-bit	LS
Windows 10	64-bit	CS
Windows Server 2008	32-bit, 64-bit	CS
Windows Server 2008 (Server Core Installation)	64-bit	LS
Windows Server 2008 R2	64-bit	CS
Windows Server 2008 R2 (Server Core Installation)	64-bit	CS
Windows Server 2012	64-bit	CS
Windows Server 2012 (Server Core Installation)	64-bit	LS
Windows Server 2012 R2	64-bit	CS
Windows Server 2016	64-bit	CS
AIX		
AIX 7.1	64-bit	CS
Debian		
Debian 6, 7	32-bit, 64-bit	CS
Debian 8	64-bit	CS
HP-UX		
HP-UX PA-RISC 11i v1		CS
HP-UX PA-RISC 11i v2 1, 11i v3		LS
HP-UX Itanium 11i v2	64-bit	LS
HP-UX Itanium 11i v3	64-bit	CS
Oracle Hardened Linux		
Oracle Hardened Linux 5.10, 6.4	32-bit, 64-bit	CS
Oracle Hardened Linux 7	64-bit	CS
Solaris		
Solaris SPARC 9, 10, 11	64-bit	CS
Solaris x86 10, 11 Intel-based	64-bit	CS
Red Hat Enterprise Linux/CentOS		
Red Hat Enterprise Linux 5, 6/CentOS 5, 6	32-bit, 64-bit	CS
Red Hat Enterprise Linux 7/CentOS 7	64-bit	LS
Red Hat Linux		
Red Hat Linux 9	32-bit	CS

LogRhythm System Monitor OS Support Levels		
Operating System	**32-bit/64-bit**	**Support Level**
SUSE		
SUSE Linux Enterprise Server 9, 11, 11.1, 12, 13	64-bit	CS
Ubuntu		
Ubuntu 12, 14, 16	64-bit	CS

Table 9 - LogRhythm System Monitor OS Support Level

The LogRhythm All-In-One XM appliances with LogRhythm Software can be used in a LogRhythm deployment instead of individual instances of the PM/DP/DX/AIE appliances.

Appliance Series	Hardware Specifications	Operating System
LR-XM4401 Series (combined PM/DP/DX/AIE server)	1 x 2.4 GHz 6 Core CPU (14.4 GHz total) 12 vCPU Dedicated Disk Drives (DAS or SAN) 64 GB RAM PERC H730 Integrated RAID Controller 4 x 1 Gigabit Ethernet NICs	Windows 2012 R2 x64 Standard Edition
LR-XM6401 Series (combined PM/DP/DX/AIE server)	2 x 2.4 GHz 6 Core CPU (28.8 GHz total) 24 vCPU Dedicated Disk Drives (DAS or SAN) 128 GB RAM PERC H730P Integrated RAID Controller 4 x 1 Gigabit Ethernet NICs	Windows 2012 R2 x64 Standard Edition
LR-XM8400 Series (combined PM/DP/DX/AIE server)	2 x 2.6 GHz 14 Core CPU (72.8 GHz total) 56 vCPU Dedicated Disk Drives (DAS or SAN) 256 GB RAM PERC H730P Integrated RAID Controller 2 x 1 Gigabit Ethernet NICs 2 x 10 Gigabit Ethernet NICs	Windows 2012 R2 x64 Standard Edition
LR-XM4500 Series (combined PM, DP, DX, AIE, Web, DC)	1 x 2.2 GHz 10 Core CPU 20 vCPU 96 GB RAM PERC H740 Integrated RAID Controller with 8GB Cache l 2 x 10 Gb/s NICs l 2 x 1 Gb/s NICs	Windows 2016 x64 Standard Edition

Appliance Series	Hardware Specifications	Operating System
LR-XM6500 Series (combined PM, DP, DX, AIE, Web, DC)	2 x 2.2 GHz 10 Core CPU 40 vCPU 196 GB RAM PERC H740 Integrated RAID Controller with 8GB Cache l 2 x 10 Gb/s NICs l 2 x 1 Gb/s NICs	Windows 2016 x64 Standard Edition
LR-XM8500 Series (combined PM, DP, DX, AIE, Web, DC)	2 x 3.0 GHz 12 Core CPU 48 vCPU 256 GB RAM PERC H740 Integrated RAID Controller with 8GB Cache l 2 x 10 Gb/s NICs	Windows 2016 x64 Standard Edition

Table 10 - LogRhythm All-In-One XM Appliances

The LogRhythm All-In-One XM software can also be installed on a customer provided platform that meets the above hardware requirements and has the following software installed.

- Windows Server 2012 R2 x64 Standard or Enterprise Edition, or
- Windows Server 2016
- Microsoft .NET Framework 4.5.2 or later
- Microsoft SQL Server 2016 SP1 Standard Edition

The LogRhythm Software can also be deployed in a virtual environment. The following table identifies the operational environment requirements.

Platform Reference	Virtual Hardware Specifications	Operating System
LR-XMVS (combined PM/DP/DX virtual server)	8 vCPU 32 GB RAM 1 NIC	Windows 2012 R2 x64 Standard Edition, or Windows Server 2016
LR-PMVS1 (dedicated PM virtual server)	4 vCPU 16 GB RAM 1 NIC	Windows 2012 R2 x64 Standard Edition, or Windows Server 2016
LR-DPVS1 (dedicated DP virtual server)	4 vCPU 8 GB RAM 1 NIC	Windows 2012 R2 x64 Standard Edition, or Windows Server 2016
LR-DXVS1 (dedicated DX virtual server)	4 vCPU 8 GB RAM 1 NIC	Windows 2012 R2 x64 Standard Edition, or Windows Server 2016

Table 11 – Virtual Hardware Specifications

2.2.1.2 Services and Products in the Operational Environment

The TOE requires an NTP Server in the operational environment to ensure time is synchronized among the distributed components. The product provides time stamps for its own use derived from the system clock managed by the underlying operating system.

2.2.2 Logical Boundaries

This section summarizes the security functions provided by the TOE:

- Security audit
- Identification and authentication
- Security management
- Protection of the TSF
- SEM Component requirements

2.2.2.1 Security audit

The TOE can generate audit records of the following security-relevant events:

- 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 the following information in each audit record it generates: 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. The Web Console records the use of identification and authentication mechanisms. 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 filter 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 LogRhythm Console or Web 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 LogRhythm Console provides deployment administration and user interaction with LogRhythm with a Graphical User Interface (GUI). The console provides the capability to manage the auditing, analysis and reaction functions. The Deployment Manager is a utility window in the LogRhythm Console. People with LogRhythm administrator credentials use it to configure and manage LogRhythm components and functionality such as alarming and reporting. The management functions are restricted to administrative roles.

The TOE comes with the following pre-defined security roles: Global Administrator, Restricted Administrator, Global Analyst, and Restricted Analyst. 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

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 TLS/HTTPS to protect data transmitted between the TOE components from unauthorized disclosure and modification. The TOE supports both self-signed certificates and user-supplied certificates for establishing TLS-protected communication. All TLS/HTTPS functionality is provided by the operating system in the operational environment.

The TOE Establishes secure communication channels between physically distributed components using TDS over TLS (for SQL Server Clients) or TLS (Mediator/System Monitor Agent, AIComMgr/Mediator, Client Console, Web Console).

Timestamps are provided by the operational environment. The TOE normalizes time stamps to account for time zone differences.

2.2.2.5 Security Event Manager Component Requirements

The System Monitors are able to collect logs from multiple sources. The TOE analyzes the collected logs and performs correlation, pattern recognition, classification assignment, the processing of metadata and event identification. The TOE 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 Client and Web Console. A potential loss of logs is prevented by the layered architecture of the TOE's solution and by providing administrative interfaces to configure allocated storage and available disk storage.

2.2.3 Excluded Product Functionality

The following features and capabilities of the TOE described in the guidance documentation are not included within the scope of the evaluation:

- User Activity Monitor (UAM)
- Data Loss Defender
- File Integrity Monitor
- High Availability
- LogRhythm Backup and Recovery Procedures
- Performance Counters
- Log Processing Report
- Network Visualization
- Save Investigation as a Report
- Reporting Center
- Customizing Reports
- Web Console Single Sign-on (introduced in Release 7.6.0)
- RESTful Alarm API (introduced in Release 7.7.0)

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 are not within the scope of evaluation.

2.3 TOE Documentation

LogRhythm has a number of administration and configuration guides for the LogRhythm Integrated Solution which include the following:

- LogRhythm NextGen SIEM 7.8.0 Help, September 14, 2021
- LogRhythm System Monitor Compatibility and Functionality Guide, Version 7.8.0, May 4, 2022
- LogRhythm Release Notes Version 7.8.0 GA, September 13, 2021
- LogRhythm Install a New LogRhythm Deployment, Version 7.8.0, August 9, 2021
- LogRhythm Web Console User Guide, Version 7.8.0, September 14, 2021
- LogRhythm Upgrade a LogRhythm Deployment, Version 7.8.0, September 13, 2021
- LogRhythm Message Processing Editor Rule Builder Guide, Version 7.8.0, September 13, 2021
- LogRhythm Schema Dictionary and Guide, Version 7.8.0, September 13, 2021.

3. Security Problem Definition

This section defines the security problem to be addressed by the TOE, in terms of threats to be countered by the TOE or its operational environment, and assumptions about the intended operational environment of the TOE.

This section identifies assumptions as A.assumption and threats as T.threat.

3.1 Assumptions

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

Assumption	Description
A.PROTECT	The TOE software critical to security policy enforcement will be protected from unauthorized physical modification.
A.PLATFORM	The underlying operating system of each TOE component will protect the component and its configuration from unauthorized access.
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.

Table 12 – Assumptions

3.2 Threats

The following are threats identified for the TOE and the IT System the TOE monitors.

Threat	Description
T.INTEGRITY_COMPROMISE	An unauthorized user may attempt to modify or destroy audit or SEM data, thus removing evidence of unauthorized or malicious activity.
T.NO_ACCOUNTABILITY	Authorized users of the TOE perform adverse actions on the TOE, or attempt to perform unauthorized actions, which go undetected.
T.UNAUTHORIZED_ACCESS	An unauthorized user may gain access to the TOE security functions and data.
T.UNAUTHORIZED_ACTIVITY	Authorized users perform unauthorized actions on the TOE.
T.UNDETECTED_THREATS	Events generated by entities in the IT system indicative of misuse or unauthorized or malicious activity go undetected.

Table 13 - Threats

4. Security Objectives

This section identifies the security objectives of the TOE and its operational environment. The security objectives identify the responsibilities of the TOE and its environment in addressing the security problem defined in Section 3.

4.1 Security Objectives for the TOE

The following are the TOE security objectives.

Objective	Description
O.ANALYZER	The TOE shall analyze collected SEM data in order to identify misuse and unauthorized or malicious activity and shall be able to record the results of its analysis.
O.AUDIT	The TOE shall be able to generate audit records of security-relevant events.
O.AUDIT_REVIEW	The TOE shall provide a means for authorized users to review the audit records generated by the TOE.
O.I_AND_A	The TOE shall provide a means for users to be identified and authenticated before gaining access to TOE services.
O.INTEGR	The TOE must ensure the integrity of all audit and SEM data
O.RESPONSE	The TOE shall respond to misuse and unauthorized or malicious activity it identifies based on its configuration.
O.REVIEW	The TOE shall provide capabilities for effective review of stored SEM data.
O.SECURITY_MANAGEMENT	The TOE shall restrict the ability to perform security management functions on the TOE to authorized administrators having appropriate privileges.
O.SENSORS	The TOE shall collect SEM data from the IT system the TOE is monitoring and to provide that SEM data to the TOE in a form suitable for the TOE to analyze.
O.STORAGE	The TOE shall protect stored audit records and SEM data from unauthorized modification or deletion.

Table 14 - Security Objectives for the TOE

4.2 Security Objectives for the Operational Environment

The security objectives for the operational environment are addressed below:

OBJECTIVE	DESCRIPTION
OE.PERSONNEL	Those responsible for the TOE must ensure that personnel working as authorized administrators have been carefully selected and trained for proper operation of the TOE.
OE.PHYSICAL	Those responsible for the TOE must ensure that those parts of the TOE critical to security policy are protected from any physical attack.
OE.PLATFORM	The underlying operating system of each TOE component will protect the component and its configuration from unauthorized access.
OE.TIME	The underlying operating system of the TOE provides a reliable time source for use by the TOE.
OE.CONFID	Those responsible for the TOE must ensure the operational environment provides capabilities to protect the confidentiality of data communicated by the TOE components and the administrative users to the TOE.

Table 15 – Security Objectives for the Operational Environment

5. IT Security Requirements

The security requirements for the TOE have been drawn from Parts 2 and 3 of the Common Criteria. The security functional requirements have been selected to correspond to the actual security functions implemented by the TOE while the assurance requirements have been selected to offer a low to moderate degree of assurance that those security functions are properly realized.

5.1 Extended Components Definition

5.1.1 Security Event Manager Component Requirements

This ST defines a new functional class for use within this ST: Security Event Manager (SEM). This family of SEM requirements was created to specifically address the collection of remote logs and the analysis by a SEM. The audit family of the CC (FAU) was used as a model for creating these requirements. The purpose of this family of requirements is to address the unique nature of SEM data and specify requirements for collecting, analyzing and reviewing SEM data.

5.1.1.1 SEM Data Collection (SEM_LDC)

This family defines requirements for being able to collect log data from targeted IT resources.

Management: SEM_LDC_EXT.1

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

a) maintenance of the parameters that control SEM data collection.

Audit: SEM_LDC_EXT.1

There are no auditable events foreseen.

SEM_LDC_EXT.1: SEM Data Collection

Hierarchical to: No other components.

Dependencies: None

SEM_LDC_EXT.1.1 The TSF shall be able to collect raw log information from remote targeted IT System resources.

Application Note: *The ST will define the log collection capabilities of the TOE. The ST will identify all compatible logs in which the TOE is able to collect and forward for processing.*

SEM_LDC_EXT.1.2 At a minimum, the TSF shall collect and record the following information:

a) Remote log records
b) Log source name
c) Log source type
d) Date and time of the log

Application Note: *The security target will identify the different log source types such as syslog, NetFlow, sFlow, SNMP traps, etc.*

5.1.1.2 SEM Analyzer (SEM_ANL)

This family defines requirements for being able to analyze collected SEM data.

Management: SEM_ANL_EXT.1

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

a) maintenance of the parameters that control SEM data analysis.

Audit: SEM_ANL_EXT.1

There are no auditable events foreseen.

SEM_ANL_EXT.1: SEM Analyzer analysis

Hierarchical to: No other components.

Dependencies: SEM_LDC_EXT.1

SEM_ANL_EXT.1.1 The TSF shall perform the following analysis function(s) on all SEM data received:
a) **[selection: *log source host, log source type, correlation, pattern recognition, assign a classification, identify events, process metadata*]**; and
b) **[assignment: *other analytical functions*]**.

Application Note: *Classification is the grouping of log messages into logical containers. Log source host is the system identification of the log source. Some logs are more important to the organization's operation, security, and compliance than others. In the identification process, the more important logs are designated as events. Log source type is the different log source types such as syslog, NetFlow, sFlow, SNMP traps, etc. Correlation is a search based upon the field values within the collected log data. Pattern recognition is the identification of a specific pattern in the collected log data to isolate interesting pieces of information.*

SEM_ANL_EXT.1.2 The TSF 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, processed metadata fields; and
b) **[assignment: *other security relevant information about the result*]**.

Application Note:

Classification is the grouping of log messages into logical containers. The processed metadata fields can include information from the original log or log-derived data where the value of the field is not part of the original log.

5.1.1.3 SEM Reaction (SEM_RCT)

This family defines requirements for being able to react to the results of SEM data analysis.

Management: SEM_RCT_EXT.1

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

a) maintenance of the parameters that control SEM reaction.

Audit: SEM_RCT_EXT.1

There are no auditable events foreseen.

SEM_RCT_EXT.1: SEM reaction

Hierarchical to: No other components.

Dependencies: SEM_ANL_EXT.1

SEM_RCT_EXT.1.1 The TSF shall send an alarm to **[assignment: *alarm destination*]** and take **[assignment: *appropriate actions*]** when a SEM event is detected.

Application Note: *There must be an alarm, though the ST should refine the nature of the alarm and define its target (e.g., administrator console, audit log).*

5.1.1.4 SEM Restricted Data Review (SEM_RDR)

This family defines requirements for reviewing SEM data and restricting access to SEM data.

Management: SEM_RDR_EXT.1

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

b) maintenance of the group of users with read access rights to the SEM data.

Audit: SEM_RDR_EXT.1

There are no auditable events foreseen.

SEM_RDR_EXT.1: SEM restricted data review

Hierarchical to: No other components.

Dependencies: SEM_LDC_EXT.1, SEM_ANL_EXT.1

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

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

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

Application Note: *This requirement applies to authorized users of the TOE. The requirement is left open for the writers of the ST to define which authorized users may access what* SEM *data.*

5.1.1.5 SEM Data Storage (SEM_STG)

This family defines requirements for securely storing SEM data.

Management: SEM_STG_EXT.1, SEM_STG_EXT.2

There are no management actions foreseen.

Audit: SEM_STG_EXT.1, SEM_STG_EXT.2

There are no auditable events foreseen.

SEM_STG_EXT.1: Guarantee of SEM data availability

Hierarchical to: No other components.

Dependencies: SEM_LDC_EXT.1, SEM_ANL_EXT.1

SEM_STG_EXT.1.1 The TSF shall protect the stored SEM data from unauthorized deletion.

SEM_ STG_EXT.1.2 The TSF shall protect the stored SEM data from modification.

Application Note: *Authorized deletion of data is not considered a modification of* SEM *data in this context. This requirement applies to the actual content of the* SEM *data, which should be protected from any modifications.*

SEM_ STG_EXT.1.3 The TSF shall ensure that **[assignment: *metric for saving SEM data*]** SEM data will be maintained when the following conditions occur: **[selection: *SEM data storage exhaustion, failure, attack***].

Application Note: *The ST needs to define the amount of* SEM *data that could be lost under the identified scenarios.*

SEM_STG_EXT.2: Prevention of SEM data loss

Hierarchical to: No other components.

Dependencies: SEM _STG_EXT.1

SEM _STG_EXT.2.1 The TSF shall **[selection: *ignore SEM data, overwrite the oldest stored SEM data, delete the oldest SEM data***] and send an alarm if the SEM storage capacity has been reached.

Application Note: *The ST must define what actions the TOE takes if the storage capacity has been reached. Anything that causes the TOE to stop collecting and analyzing SEM data may not be the best solution, as this will only affect the TOE and not the IT resource(s) the TOE is monitoring.*

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 5 and the extended components defined in Section 5.1.

The functional security requirements for this Security Target consist of the following components from Part 2 of the CC, of which are summarized in the following table:

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
SEM: Security Event Manager Component Requirements	SEM_ANL_EXT.1: SEM analyzer analysis
	SEM_RCT_EXT.1: SEM reaction
	SEM_RDR_EXT.1: SEM restricted data review
	SEM_STG_EXT.1(1): Guarantee of SEM data availability – General SEM_STG_EXT.1(2): Guarantee of SEM data availability – AI Engine Server SEM_STG_EXT.1(3): Guarantee of SEM data availability – Data Indexer
	SEM_STG_EXT.2(1): Prevention of SEM data loss – General SEM_STG_EXT.2(2): Prevention of SEM data loss – AI Engine Server SEM_STG_EXT.2(3): Prevention of SEM data loss – Data Indexer
	SEM_LDC_EXT.1: SEM Data Collection

Table 16 - 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 [***not specified***] level of audit; and

c) [Access to the TOE and SEM data]

FAU_GEN.1.2 The TSF shall record within each audit record at last 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 17 Auditable Events]**.

Component	Event	Details
FAU_GEN.1	Start-up and shutdown of audit functions	
FAU_GEN.1	Access to the TOE and SEM 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 17 – Auditable Events

5.2.1.2 Audit Review (FAU_SAR.1)

FAU_SAR.1.1 The TSF shall provide **[Global 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)

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)

FAU_SAR.3.1 The TSF shall provide the ability to apply [**filtering**] 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**] stored 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)

FAU_STG.4.1 The TSF shall [***prevent audited events*****]** 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.1)

FIA_ATD.1.1 The TSF shall maintain the following list of security attributes belonging to individual users: [
a) User identity;
b) Authentication data;
c) Authorizations].

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 SEM data collection, analysis and reaction**] to [**Global Administrators and Restricted Administrators**].

5.2.3.2 Management of TSF Data (FMT_MTD.1)

FMT_MTD.1.1 The TSF shall restrict the ability to [***query, modify*****]** the **[set of audited events, user accounts]** to [**Global Administrators]**.

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 SEM data collection, analysis and reaction**].

5.2.3.4 Security Roles (FMT_SMR.1)

FMT_SMR.1.1 The TSF shall maintain the roles

[**Global Administrator,**

Global Analyst,

Restricted Administrator,

Restricted Analyst].

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

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 Security Event Manager Component Requirements

5.2.5.1 SEM Data Collection (SEM_LDC_EXT.1)

SEM_LDC_EXT.1.1 The TSF shall be able to collect raw log information from remote targeted IT System resources.

SEM_LDC_EXT.1.2 At a minimum, the TSF shall collect and record the following information:

a) Remote log records
b) Log source name
c) Log source type
d) Date and time of the log.

5.2.5.2 SEM Analyser Analysis (SEM_ANL_EXT.1)

SEM_ANL_EXT.1.1 The TSF shall perform the following analysis function(s) on all SEM data received:

a) [***correlation, pattern recognition, assign a classification, identify events, process metadata***]; and
b) [**none**].

SEM_ANL_EXT.1.2 The TSF 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, processed metadata fields; and
b) [**SEM classification as shown in the table below**].

Classification	Sub-Classification
Audit	Startup and Shutdown
	Configuration
	Policy
	Account Created
	Account Modified
	Account Deleted
	Access Granted
	Access Revoked

Classification	Sub-Classification
	Authentication Success Authentication Failure Access Success Access Failure Other Audit Success Other Audit Failure Other Audit
Security	Compromise Attack Denial of Service Malware Suspicious Reconnaissance Misuse Activity Failed Attack Failed Denial of Service Failed Malware Failed Suspicious Failed Activity Other Security
Operations	Critical Error Warning Information Network Allow Network Deny Network Traffic Other Operations

Application Note:

Classification is the grouping of log messages into logical containers. The processed metadata fields can include information from the original log or log-derived data where the value of the field is not part of the original log.

5.2.5.3 SEM Reaction (SEM_RCT_EXT.1)

SEM_RCT_EXT.1.1 The TSF shall send an alarm to [**the alarm database and users configured to receive alarms**] and take [**configured remediation actions**] when a SEM event is detected.

5.2.5.4 SEM Restricted Data Review (SEM_RDR_EXT.1)

SEM_RDR_EXT.1.1 The TSF shall provide [**Global Administrators, Global Analysts, Restricted Administrators, and Restricted Analysts**] with the capability to read [**all analyzer data that the administrator is authorized to view**] from the SEM data.

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

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

5.2.5.5 Guarantee of SEM Data Availability – General (SEM_STG_EXT.1(1))

SEM_STG_EXT.1.1(1) The TSF shall protect the stored SEM data from unauthorized deletion.

SEM_STG_EXT.1.2(1) The TSF shall protect the stored SEM data from modification.

SEM_STG_EXT.1.3(1) The ~~TSF~~ **System Monitor, Data Processor, and Platform Manager components** shall ensure that [**100% of existing**] SEM data will be maintained when the following conditions occur: [***SEM data storage exhaustion***].

Application Note: The TOE is designed as a distributed system. This ST iterates the SEM data storage requirements to specify behavior applicable to each component.

5.2.5.6 Guarantee of SEM Data Availability – AI Engine Server (SEM_STG_EXT.1(2))

SEM_STG_EXT.1.1(2) The TSF shall protect the stored SEM data from unauthorized deletion.

SEM_STG_EXT.1.2(2) The TSF shall protect the stored SEM data from modification.

SEM_STG_EXT.1.3(2) The ~~TSF~~ **AI Engine Server** shall ensure that [**a block with Global administrator-configured size of**] SEM data will be maintained when the following conditions occur: [***SEM data storage exhaustion***].

5.2.5.7 Guarantee of SEM Data Availability – Data Indexer (SEM_STG_EXT.1(3))

SEM_STG_EXT.1.1(3) The TSF shall protect the stored SEM data from unauthorized deletion.

SEM_STG_EXT.1.2(3) The TSF shall protect the stored SEM data from modification.

SEM_STG_EXT.1.3(3) The ~~TSF~~ **Data Indexer** shall ensure that [**the most recent 80%**] SEM data will be maintained when the following conditions occur: [***SEM data storage exhaustion***].

5.2.5.8 Prevention of SEM Data Loss – General (SEM_STG_EXT.2(1))

SEM_STG_EXT.2.1(1) The ~~TSF~~ **System Monitor, Data Processor, and Platform Manager** shall [***ignore SEM data***] and send an alarm if the SEM storage capacity has been reached.

5.2.5.9 SEM _STG_EXT.2(2) Prevention of SEM Data Loss – AI Engine Server (SEM _STG_EXT.2(2))

SEM_STG_EXT.2.1(2) The ~~TSF~~ **AI Engine Server** shall [***overwrite the oldest stored SEM data***] and send an alarm if the SEM storage capacity has been reached.

5.2.5.10 SEM _STG_EXT.2(2) Prevention of SEM Data Loss – Data Indexer (SEM_STG_EXT.2(3))

SEM_STG_EXT.2.1(3) The ~~TSF~~ **Data Indexer** shall [***delete the oldest SEM data***] and send an alarm if the SEM storage capacity has been reached.

5.3 TOE Security Assurance Requirements

The assurance security requirements for this Security Target are taken from Part 3 of the CC. These assurance requirements compose an Evaluation Assurance Level 2 augmented with ALC_FLR.2 (EAL2+). The assurance components are summarized in the following table:

Requirement Class	Requirement Component
ADV: Development	ADV_ARC.1: Security architecture description
	ADV_FSP.2: Security-enforcing functional specification
	ADV_TDS.1: Basic design
AGD: Guidance documents	AGD_OPE.1: Operational user guidance
	AGD_PRE.1: Preparative procedures
ALC: Life-cycle support	ALC_CMC.2: Use of a CM system
	ALC_CMS.2 Parts of the TOE CM coverage
	ALC_DEL.1: Delivery procedures
	ALC_FLR.2: Flaw reporting procedures
ASE: Security Target evaluation	ASE_CCL.1 Conformance claims
	ASE_ECD.1 Extended components definition
	ASE_INT.1 ST introduction
	ASE_OBJ.2 Security objectives
	ASE_REQ.2 Derived security requirements
	ASE_SPD.1 Security problem definition
	ASE_TSS.1 TOE summary specification
ATE: Tests	ATE_COV.1: Evidence of coverage
	ATE_FUN.1: Functional testing
	ATE_IND.2: Independent testing — sample
AVA: Vulnerability assessment	AVA_VAN.2 Vulnerability analysis

Table 18 - EAL2 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 each SFR-enforcing TSFI, the functional specification shall describe the SFR-enforcing actions associated with the TSFI.

ADV_FSP.2.5c For each SFR-enforcing TSFI, 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 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 provide the behaviour summary of each SFR-supporting or SFR-non-interfering TSF subsystem.

ADV_TDS.1.4c The design shall summarise the SFR-enforcing behaviour of the SFR-enforcing subsystems.

ADV_TDS.1.5c The design shall provide a description of the interactions among SFR-enforcing subsystems of theTSF, 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 TSFIs trace to the behaviour described in the TOE design that they invoke.

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 fulfil 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 labelled 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.2.1e 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 and provide 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 Security Target Evaluation (ASE)

5.3.4.1 Conformance claims (ASE_CCL.1)

ASE_CCL.1.1D The developer shall provide a conformance claim.

ASE_CCL.1.2D The developer shall provide a conformance claim rationale.

ASE_CCL.1.1C The conformance claim shall contain a CC conformance claim that identifies the version of the CC to which the ST and the TOE claim conformance.

ASE_CCL.1.2C The CC conformance claim shall describe the conformance of the ST to CC Part 2 as either CC Part 2 conformant or CC Part 2 extended.

ASE_CCL.1.3C The CC conformance claim shall describe the conformance of the ST to CC Part 3 as either CC Part 3 conformant or CC Part 3 extended.

ASE_CCL.1.4C The CC conformance claim shall be consistent with the extended components definition.

ASE_CCL.1.5C The conformance claim shall identify all PPs and security requirement packages to which the ST claims conformance.

ASE_CCL.1.6C The conformance claim shall describe any conformance of the ST to a package as either package-conformant or package-augmented.

ASE_CCL.1.7C The conformance claim rationale shall demonstrate that the TOE type is consistent with the TOE type in the PPs for which conformance is being claimed.

ASE_CCL.1.8C The conformance claim rationale shall demonstrate that the statement of the security problem definition is consistent with the statement of the security problem definition in the PPs for which conformance is being claimed.

ASE_CCL.1.9C The conformance claim rationale shall demonstrate that the statement of security objectives is consistent with the statement of security objectives in the PPs for which conformance is being claimed.

ASE_CCL.1.10C The conformance claim rationale shall demonstrate that the statement of security requirements is consistent with the statement of security requirements in the PPs for which conformance is being claimed.

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

5.3.4.2 Extended components definition (ASE_ECD.1)

ASE_ECD.1.1D The developer shall provide a statement of security requirements.

ASE_ECD.1.2D The developer shall provide an extended components definition.

ASE_ECD.1.1C The statement of security requirements shall identify all extended security requirements.

ASE_ECD.1.2C The extended components definition shall define an extended component for each extended security requirement.

ASE_ECD.1.3C The extended components definition shall describe how each extended component is related to the existing CC components, families, and classes.

ASE_ECD.1.4C The extended components definition shall use the existing CC components, families, classes, and methodology as a model for presentation.

ASE_ECD.1.5C The extended components shall consist of measurable and objective elements such that conformance or nonconformance to these elements can be demonstrated.

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

ASE_ECD.1.2E The evaluator shall confirm that no extended component can be clearly expressed using existing components

5.3.4.3 ST introduction (ASE_INT.1)

ASE_INT.1.1D The developer shall provide an ST introduction.

ASE_INT.1.1C The ST introduction shall contain an ST reference, a TOE reference, a TOE overview and a TOE description.

ASE_INT.1.2C The ST reference shall uniquely identify the ST.

ASE_INT.1.3C The TOE reference shall uniquely identify the TOE.

ASE_INT.1.4C The TOE overview shall summarise the usage and major security features of the TOE.

ASE_INT.1.5C The TOE overview shall identify the TOE type.

ASE_INT.1.6C The TOE overview shall identify any non-TOE hardware/software/firmware required by the TOE.

ASE_INT.1.7C The TOE description shall describe the physical scope of the TOE.

ASE_INT.1.8C The TOE description shall describe the logical scope of the TOE.

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

ASE_INT.1.2E The evaluator shall confirm that the TOE reference, the TOE overview, and the TOE description are consistent with each other.

5.3.4.4 Security Objectives (ASE_OBJ.2)

ASE_OBJ.2.1D The developer shall provide a statement of security objectives.

ASE_OBJ.2.2D The developer shall provide a security objectives rationale.

ASE_OBJ.2.1C The statement of security objectives shall describe the security objectives for the TOE and the security objectives for the operational environment.

ASE_OBJ.2.2C The security objectives rationale shall trace each security objective for the TOE back to threats countered by that security objective and OSPs enforced by that security objective.

ASE_OBJ.2.3C The security objectives rationale shall trace each security objective for the operational environment back to threats countered by that security objective, OSPs enforced by that security objective, and assumptions upheld by that security objective.

ASE_OBJ.2.4C The security objectives rationale shall demonstrate that the security objectives counter all threats.

ASE_OBJ.2.5C The security objectives rationale shall demonstrate that the security objectives enforce all OSPs.

ASE_OBJ.2.6C The security objectives rationale shall demonstrate that the security objectives for the operational environment uphold all assumptions.

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

5.3.4.5 Derived security requirements (ASE_REQ.2)

ASE_REQ.2.1D	The developer shall provide a statement of security requirements.
ASE_REQ.2.2D	The developer shall provide a security requirements rationale.
ASE_REQ.2.1C	The statement of security requirements shall describe the SFRs and the SARs.
ASE_REQ.2.2C	All subjects, objects, operations, security attributes, external entities and other terms that are used in the SFRs and the SARs shall be defined.
ASE_REQ.2.3C	The statement of security requirements shall identify all operations on the security requirements.
ASE_REQ.2.4C	All operations shall be performed correctly.
ASE_REQ.2.5C	Each dependency of the security requirements shall either be satisfied, or the security requirements rationale shall justify the dependency not being satisfied.
ASE_REQ.2.6C	The security requirements rationale shall trace each SFR back to the security objectives for the TOE.
ASE_REQ.2.7C	The security requirements rationale shall demonstrate that the SFRs meet all security objectives for the TOE.
ASE_REQ.2.8C	The security requirements rationale shall explain why the SARs were chosen.
ASE_REQ.2.9C	The statement of security requirements shall be internally consistent.
ASE_REQ.2.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.4.6 Security Problem Definition (ASE_SPD.1)

ASE_SPD.1.1D	The developer shall provide a security problem definition.
ASE_SPD.1.1C	The security problem definition shall describe the threats.
ASE_SPD.1.2C	All threats shall be described in terms of a threat agent, an asset, and an adverse action.
ASE_SPD.1.3C	The security problem definition shall describe the OSPs.
ASE_SPD.1.4C	The security problem definition shall describe the assumptions about the operational environment of the TOE.
ASE_SPD.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

5.3.4.7 TOE Summary Specification (ASE_TSS.1)

ASE_TSS.1.1D	The developer shall provide a TOE summary specification.
ASE_TSS.1.1C	The TOE summary specification shall describe how the TOE meets each SFR.
ASE_TSS.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
ASE_TSS.1.2E	The evaluator shall confirm that the TOE summary specification is consistent with the TOE overview and the TOE description.

5.3.5 Tests (ATE)

5.3.5.1 Evidence 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.

5.3.5.2 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.

5.3.5.3 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.

5.3.6 Vulnerability assessment (AVA)

5.3.6.1 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

The security functions described in the following subsections fulfill the security requirements that are defined in **Section 5.2 TOE Security Functional Requirements**. The security functions performed by the TOE are as follows:

- Security Audit
- Identification and Authentication
- Security Management
- Protection of the TSF
- SEM Component Requirements

6.1 Security Audit

The TOE has an audit generation mechanism to record security events at a not specified level of audit.

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).

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 (FAU_GEN.1.2).

The TOE provides stored procedures in SQL Server Management Studio for the authorized Global administrator to read audit records from the audit trail (FAU_SAR.1). This interface is restricted to the authorized Global administrator role (FAU_SAR.2). The audit stored procedure provides flexible filtering, including the capability to filter 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).

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 audit records are stored in tables in a Microsoft SQL Server 2016 SP1 Standard Edition database, which ultimately are stored on a Windows Server 2012 R2 x64 Standard or Windows Server 2016 file system. Therefore the TOE relies on the operating environment for proper enforcement of file permission settings.

The following are the events the TOE can audit:

1. Start-up and shutdown of audit functions
2. Access to the TOE and SEM data
3. Reading of information from the audit records
4. Unsuccessful attempts to read information from the audit records
5. All modifications to the audit configuration that occur while the audit collection functions are operating
6. All use of the authentication mechanism
7. All use of the user identification mechanism

8. All modifications in the behaviour of the functions of the TSF
9. All modifications to the values of TSF data
10. 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 17**. 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 Global Administrators with the ability to read the audit records. These records are provided in a human 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 Global Administrators with the ability to filter 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 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.

6.2 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. When an authorized Global 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 Platform Manager SQL Server and creates a SQL Server login using the supplied local Windows OS or Active Directory user account.
2. The Platform Manager SQL Server stores the login in its internal tables.
3. The LogRhythm Console extracts the SQL Server login from the Platform 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 Platform Manager databases with the appropriate database role membership based on the LogRhythm user's role (Global Administrator, Global Analyst, Restricted Administrator, or Restricted Analyst).
5. The Web Console logins are associated with the same User Profiles defined in the Client Console. Each profile has specific permissions in the Web Console.

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).

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 Administrator and Restricted Analyst Administrator roles, '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.

6.3 Security Management

The TOE provides the capability to manage the auditing, analysis and reaction functions. The management functions are restricted to the permissions assigned to each user profile.

Each LogRhythm user account must be assigned a User Profile. A User Profile is assigned one of the following Security Roles: Global Administrator, Restricted Administrator, Global Analyst, or Restricted Analyst. The general permissions for each Security Role are described below.

- **Global Administrator**
  - Full control of LogRhythm functionality. The LogRhythm Global Administrator also has full Windows Administrator and SQL Server SA Account rights.
- **Global Analyst**
  - Has LogRhythm database permissions that are limited only to read-only functions (investigations, tails, alarms, and reports)
  - Has no access to Deployment Manager to make configuration changes (LogRhythm enforced)
- **Restricted Administrator**
  - Has LogRhythm database permissions that permit configuration changes as well as read-only functions (investigations, tails, alarms and reports)
  - Has specific permissions to view and modify host, System Monitors and log source properties (SQL Server enforced, View). Also known as LogRhythm Discretionary Access Control. This access control is segregated by Entity.
  - Has limited access to Deployment Manager to make configuration changes to permitted resources (LogRhythm enforced)
- **Restricted Analyst**
  - Has LogRhythm database permissions that are limited to read-only functions (investigations, tails, alarms and reports)
  - Has specific permissions to view data from log sources, entities and Data Processors (SQL Server enforced, View). Also known as LogRhythm Discretionary Access Control. This access control is primarily segregated by Entity, but can also be granted at a Data Processor and/or Log Source level.
  - Has no access to Deployment Manager to make administrative changes (LogRhythm enforced)

These roles, when assigned to users, provide varying levels of access to the TOE interfaces and functions (FMT_SMR.1).

The Global Administrator role has full control of the configuration and data. The Global Administrator has the overall responsibility of managing and configuring the TOE 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. This includes configuration of alarm rules, and archive policies and settings.

The Global Administrator can create, modify, delete, configure, download updates and implement the rules on the TOE. In addition, the Global 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 Administrator can grant users with the Restricted Administrator role access to database resources pertaining to specific Log Sources. A Restricted Administrator does not have access to all of the tabs on the Deployment Manager that a Global Administrator does. The list of tabs available to a Restricted Administrator is: Entities, System Monitor, Log Sources, and Alarm Rules.

Users must have one of the four 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 and the user accounts.
- FMT_SMF.1: The TOE provides interfaces to manage the audit generation function.
- FMT_SMR.1: The TOE maintains user role attributes. There are four pre-defined built-in roles.

6.4 Protection of the TSF

All communication channels between TOE components are protected by TLS (FPT_ITT.1).

The TOE uses TLS/HTTPS to protect data transmitted between the TOE components from unauthorized disclosure and modification. All TLS/HTTPS functionality is provided by the operating system in the operational environment.

The TOE Establishes secure communication channels between physically distributed components using TDS over TLS (for SQL Server Clients) or TLS (Mediator/System Monitor Agent, AIComMgr/Mediator, Client Console, Web Console).

TLS 1.2 is used to protect communications between all the distributed TOE components.

The TOE supports TLS v1.2 with the following ciphersuites:

- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
- TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
- TLS_DHE_RSA_WITH_AES_256_GCM_SHA384
- TLS_DHE_RSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384
- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA
- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA

- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA
- TLS_RSA_WITH_AES_256_GCM_SHA384
- TLS_RSA_WITH_AES_128_GCM_SHA256
- TLS_RSA_WITH_AES_256_CBC_SHA256
- TLS_RSA_WITH_AES_128_CBC_SHA256
- TLS_RSA_WITH_AES_256_CBC_SHA
- TLS_RSA_WITH_AES_128_CBC_SHA

Data sent between the Console, Mediator Server Service, AI Engine Service, ARM, and Job Manager and Microsoft SQL Server 2016 SP1 Standard Edition databases use 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., Client Console, Mediator Server, AI Engine, ARM, and Job Manager) and Microsoft SQL Server 2016 SP1 Standard Edition. Data sent between System Monitors and Mediator Servers and between Mediator Servers and AI Engine Communication Managers are protected using TLS services as described below. HTTPS is used to protect the authentication, configuration, and search requests between the API Gateway on the Web Console, Platform Manager, Data Indexer, Client Console, and Data Processor.

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 generates a self-signed server certificate. TLS clients use these certificates to authenticate the identity of the servers. That is, a System Monitor 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. 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 Global administrator-specified server certificates. A Global administrator can specify a Mediator Server Service server certificate, an AI Engine Service server certificate, or both. If the administrator does not specify a server certificate, the TOE defaults to a self-signed certificate as in the one-way authentication configuration. A Global administrator can specify a SQL Server TLS server certificate through the operational environment. Moreover, a Global administrator can configure each Mediator Server Service and each AI Engine Communication Manager to require a TLS client certificate. The Global 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 Monitors and an AI Engine Service will accept connections only from a list of authorized Mediator Servers.

The following communication channels are protected:

- Client Console to/from Microsoft SQL Server 2016 SP1 Standard Edition (Platform Manager) Communications

  Encryption is provided when 'Encrypt all communications' on the Client Console Login Screen has been configured. Once the Client 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 use SQL Server TDS with TLS encryption protocols. HTTPS is used to protect the authentication, configuration, and search requests between the API Gateway on the Platform Manager and Client Console.

- Client Console to Data Indexer

  The Client Console subsystem interacts with the Data Indexer subsystem for authentication and configuration, and searching the stored machine and forensic data. All traffic between the Client Console and the Data Indexer is protected by a HTTPS channel.

- Web Console to the Platform Manager

  The Web Console SQL Server communicates via SQL TDS with TLS to access the Platform Manager Events database. HTTPS is used to protect the authentication, configuration, and search requests between the API Gateway on the Web Console and the Platform Manager.

- Web Console to the Data Indexer

  The Web Console communicates with the Data Indexer for authentication, configuration, and to request searches on the stored raw and structured logs. All communication between the Web Console and the Data Indexer is encrypted via HTTPS.

- System Monitor/Data Collector to Data Processor Communications:

  Once the System Monitor starts up, it reads the scsm.ini file found in the LogRhythm System Monitor config file to obtain the IP Address of the Mediator (on the Platform Manager). The System Monitor then initiates a connection to the Mediator Server. A Mediator Server will accept a connection only from a registered System Monitor. Registration establishes the IP address of each System Monitor. The System Monitor and Mediator Server negotiate a secure TLS communication channel as configured (for example with one-way or two-way authentication).

- Data Processor to AI Engine Communication Manager:

  When a Data Processor 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 and forwards log data to the AIE 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 TLS communication channel as configured (for example, with one-way or two-way authentication). All traffic between Mediator Servers and the Communication Manager is encrypted.

- AI Engine to Platform Manager
  The AI Engine receives configuration details from the Platform Manager EMDB via SQL Server's TDS and encryption protocols. HTTPS is used to protect the authentication, configuration, and search requests between the API Gateway on the AI Engine and the Platform Manager.

- Data Processor to Data Indexer

  The Data Processor forwards raw and structured logs to the Data Indexer. The communication channel between the Data Processor to the Data Indexer is protected from modification and disclosure by a Global administrator configured HTTPS tunnel.

- Data Processor to Platform Manager
  The Data Processor obtains SQL Server access to the Platform Manager EMDB using SQL Server TDS with TLS encryption protocols. HTTPS is used to protect the authentication, configuration, and search requests between the API Gateway on the data Processor and the Platform Manager.

- Data Indexer to Platform Manager

  The Data Indexer communicates with the Platform Manager to connect to the EMDB and read/update tables. All connections and communications are handled using SQL Server TDS with TLS encryption protocols. HTTPS is used to protect the authentication, configuration, and search requests between the API Gateway on the Platform Manager and Data Indexer.

The Mediator Server, AI Engine, ARM, Client 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 Client 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 Microsoft SQL Server 2016 SP1 Standard Edition.

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 TLS to protect data transmitted between the TOE components from unauthorized disclosure and modification.

6.5 Security Event Manager Component Requirements

The TOE monitors an IT System for activity that may inappropriately affect the IT System's assets. The System Monitor collects logs from all log sources including syslog, snmp, netflow and sflow devices, Windows events, flat file, databases or applications. The TOE will collect at a minimum the following information: log source name, log source type, and the date and time of the log (SEM_LDC_EXT.1). In addition, some logs contain location, service, protocol information; source and destination addresses; and other information specific to the type of log collected.

Information is gathered by the TOE System Monitors and sent to a Data Processor (Mediator Server) which analyzes the data against defined rules. The TOE performs correlation, pattern recognition, assignment classification, metadata processing, and the identification of events on all log files received (SEM_ANL_EXT.1).

The Client Console and Web Console provide 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. During analysis, the Mediator Server compares the log against the defined knowledge-based rules to determine if the log should be forwarded to the Platform Manager as an 'Event', where it is stored and actions can be taken as a result.

The LogRhythm Data Processor (Mediator Server) log processing (SEM_ANL_EXT.1) includes the following:

- Initial Processing
  - Assign a Log Source - Log Sources are unique log originators on a specific Host. LogRhythm identifies the Log Source to:
    - Determine where the log originated.
    - Assign the Log Source to the correct set of base rules called a Message Processing Engine (MPE) policy.
    - Process the log against the assigned MPE policy.
  - Identify a Common Event - Processing the log against the appropriate rule base identifies the Common Event which is a short, plain-language description of the log.
  - Assign a Classification - Based on the Common Event, the TOE can assign a classification to categorize logs and events. The TOE includes three major classifications – Operations, Audit, or Security – and a more specific sub-classification as identified in the following table:

Classification	Sub-Classification
Audit	Startup and Shutdown
	Configuration
	Policy
	Account Created
	Account Modified
	Account Deleted
	Access Granted
	Access Revoked
	Authentication Success
	Authentication Failure

Classification	Sub-Classification
	Access Success Access Failure Other Audit Success Other Audit Failure Other Audit
Security	Compromise Attack Denial of Service Malware Suspicious Reconnaissance Misuse Activity Failed Attack Failed Denial of Service Failed Malware Failed Suspicious Failed Activity Other Security
Operations	Critical Error Warning Information Network Allow Network Deny Network Traffic Other Operations

Table 19 – LogRhythm Data Classifications

  - Identify Events - LogRhythm recognizes that some logs are more important to the organization's operation, security, and compliance than others. In the identification process, the more important logs are designated as Events.

- Metadata Processing - LogRhythm parses, calculates, and derives metadata from logs. The metadata fields go into a database to help speed performance when the LogRhythm search tools are used.
  - Parsed data – the items are parsed from a raw log.
    These meta data fields include: IANA Protocol Number, IANA Protocol Name , Object, Object Name, Object Type, Hash, Policy, Result, URL, User Agent, Response Code, Subject, Version, Command, Reason, Action, Status, Session Type, Process Name, Process ID, Parent Process ID, Parent Process Name, Parent Process Path, Quantity, Amount, Size, Rate, Session, Severity, Vendor Message ID, Vendor Info, Threat Name, Threat ID, CVE, Origin MAC Address, Impacted MAC Address, Origin Interface, Impacted Interface, IP Address (Origin), SIPv4, SIPv6, SIPv6E, Origin Hostname, Origin Hostname or IP, Origin NAT IP, DIP/DestinationIP/Impacted IP, DIPv4, DIPv6, DIPv6E, Impacted Hostname, Impacted Hostname or IP, Impacted NAT IP, Serial Number, Login → User (Origin), Account → User (Impacted), Sender, Recipient, Group, Entity (Origin), Entity (Impacted), Zone (Origin), Zone (Impacted), Location (Origin), Location

(Impacted), Country (Origin), Domain, Origin Port, Impacted Port, Origin NAT Port, Impacted NAT Port.

- Calculated - These fields are calculated from source data that is parsed in fields such as Host (Impacted) Kbytes Rcvd, Host (Impacted) Kbytes Sent, and Host (Impacted) Kbytes Total.
- Derived data – LogRhythm derived metadata fields store network and host information pulled from the log message. The derived data is not parsed in the schema, but is instead inferred and built from other metadata fields

The processed metadata fields can include information from the original log or log-derived data where the value of the field is not part of the original log and can consist of the following: Application, Known Application, Duration, Classification, Common Event, Priority, Direction, MPE Rule Name, Host (Origin), Host (Impacted), Known Host (Origin), Entity (Origin), Entity (Impacted), Zone (Origin), Zone (Impacted), Location (Origin), Location (Impacted), Country (Origin), Country (Impacted), Log Date/Normal Date, Log Count, Log Source Entity, Log Source Type, Log Source Host, Log Source, Log Sequence Number, Log Message, First Log Date, Last Log Date, Network (Origin), Network (Impacted), User Identity (Origin), User Identity (Impacted), Recipient Identity, Sender Identity.

A Data Processor (Mediator Server) may send log metadata to an AI Engine Server for additional analysis of sets of log messages over time (SEM_ANL_EXT.1). The analysis process includes identifying and categorizing the log messages and determining if they will be forwarded to the Platform Manager as an Event.

The LogRhythm ARM service, which resides on the Platform Manager, 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. A Global 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. It uses a plug-in architecture that executes scripts on a Windows system in a scripting language such as power shell (SEM_RCT_EXT.1). Authorized administrators can view, and work with alarm notification policies via the My LogRhythm menu on the Client 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 Administrator, Global Analysts, Restricted Administrators, and Restricted Analyst administrative roles. The logs and Events are provided in a readable format to authorized users (SEM_RDR_EXT.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.

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 access to the logs. These interfaces are restricted to users with the authorized Global Admin administrative role. (SEM_STG_EXT.1(1), SEM_STG_EXT.1(2), SEM_STG_EXT.1(3)

There are two types of resources that affect log collection when exhausted: allocated storage and available disk storage. A Global Admin allocates persistent log storage for the Data Processors and AI Engine servers. In addition, the operating system provides disk storage to meet the Global Admin's allocation. The TOE may exhaust either type of storage. A potential loss of logs is prevented by the layered architecture of the TOE's solution and by providing administrative interfaces to configure allocated storage and available disk storage.

To prevent local storage exhaustion on a Data Processor, a Global 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.

The Data Processor mediator server receives logs from the Data Collector/System Monitors and stores them in memory. If the Data Processor Message Processing Engine (MPE) receives a large volume of logs where it cannot process all the logs, memory will fill up. The excess logs are written as persistent files on the hard drive. Once the logs stored in memory are processed, the MPE reads the logs from persistent storage, processes them, and in turn

frees up the disk space. (SEM_STG_EXT.1(1)) The Data Processor stores also stores logs in archive files on the disk for possible future re-processing. The archive files can be viewed and searched by authorized administrators.

For systems with a high volume of logs, more than one Data Processor is configured to accept the incoming logs. If the Data Processor drive storage becomes full in accordance with the previously note limits, then the Data Processor Mediator Server service will go into a 'suspend' mode and stop accepting logs and incoming connections from System Monitors and the System Monitors will send the logs to another Data Processor. Once suspended, the LogRhythm Mediator Server service writes an event and displays a WARNING in a file stored on a separate volume indicating that the system is in a 'suspend' mode. If disk space is completely exhausted on a Data Processor, it will shut down and all LogRhythm components will cease to operate.

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

1. For interface types that are read by the System Monitor including flat ASCII files, Windows Event Logs, database logs (residing in a database table), and Cisco SDEE sources, the System Monitor will suspend collection relying on the logging systems to buffer logs until the System Monitor can resume collection as individual Data Processors come back online.
2. For interface types that push data to the System Monitor including syslog (UDP and TCP), Cisco NetFlow, and Checkpoint LEA interface, the System Monitor 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 will typically cease to function. Logs that are pushed to the System Monitor are at greatest risk of loss (e.g. syslog which only exists on the network until reception at a System Monitor).

The System Monitor 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 System Monitor is no longer collecting logs. The Mediator also logs an error indicating a suspend condition resulting from minimum disk space. (SEM_STG_EXT.1(1), SEM_STG_EXT.2(1)). When Data Processors come back online the System Monitor will continue sending current data and will periodically read in the data written to disk and forward to a Data Processor.

The Data Processor forwards raw and structured logs to the Data Indexer. If the Data Indexer data disk reaches 80% capacity, the oldest SEM data logs will be deleted and send an alarm (SEM_STG_EXT.1(3), SEM_STG_EXT.2(3)). The size of the Data Indexer storage capabilities should be configured according to the product administrative guides in order to provide sufficient storage space for expected logs. The Data Indexer 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.

An AI Engine Server handles SEM data storage exhaustion by deleting oldest SEM data. Each AI Engine Server has local, persistent storage where it buffers log data files it receives from Data Processors. 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 SEM data storage is exhausted (SEM_STG_EXT.1(2), SEM_STG_EXT.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 Platform Manager. These log files can be re-created by choosing the raw log data from the Data Indexer 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 Platform Manager 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. Disk space monitoring is configured and if disk space is not available, an error message is recorded in the log files. (SEM_STG_EXT.1(1), SEM_STG_EXT.2(1))

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

- SEM_LDC_EXT.1: The TOE is able to collect SEM data from the targeted IT System resources and records various details of the event.
- SEM_ANL_EXT.1: The TOE analyzes the collected logs and performs correlation, pattern recognition, classification assignment, the processing of metadata, event identification and records information about the analytical result.
- SEM_RCT_EXT.1: The TOE can send alarms to the alarm database and also to users configured to receive alarms.
- SEM_RDR_EXT.1: The TOE provides authorized administrators with the ability to read analyzer data and restricts this capability to these authorized administrators.
- SEM_STG_EXT.1(1): The TOE protects the stored data from unauthorized deletion and modification. System Monitors, Data Processors, and the Platform Manager preserve existing SEM data in the event of storage exhaustion.
- SEM_STG_EXT.1(2): The TOE protects the stored data from unauthorized deletion and modification. The AI Engine Servers preserve newest SEM data in the event of storage exhaustion.
- SEM_STG_EXT.1(3): The TOE protects the stored data from unauthorized deletion and modification. The Data Indexers preserve the most recent 80% of the SEM data in the event of storage exhaustion.
- SEM_STG_EXT.2(1): System Monitors, Data Processors and the Platform Manager TOE components ignore SEM data and send alarms if storage capacity is reached.
- SEM_STG_EXT.2(2): AI Engine Servers overwrite oldest stored SEM data and send alarms if storage capacity is reached.
- SEM_STG_EXT.2(3): Data Indexers delete the oldest stored SEM data and send alarms if storage capacity is reached.

7. 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.

7.1 Security Objectives Rationale

This section shows that all secure usage assumptions and threats are completely covered by security objectives for the TOE or operational environment. In addition, each objective counters or addresses at least one assumption or threat.

	T.INTEGRITY_COMPROMISE	T.NO_ACCOUNTABILITY	T.UNAUTHORIZED_ACCESS	T.UNAUTHORIZED_ACTIVITY	T.UNDETECTED_THREATS	A.PROTECT	A.PLATFORM	A.MANAGE	A.NOEVIL
O.ANALYZER					X				
O.AUDIT		X							
O.AUDIT_REVIEW		X							
O.I_AND_A			X						
O.INTEGR	X								
O.RESPONSE					X				
O.REVIEW					X				
O.SECURITY_MANAGEMENT				X					
O.SENSORS					X				
O.STORAGE	X								
OE.PERSONNEL								X	X
OE.PHYSICAL						X			
OE.PLATFORM			X				X		
OE.TIME		X			X				
OE.CONFID	X								

Table 20 - Security Problem Definition to Security Objective Correspondence

7.1.1.1 T.INTEGRITY_COMPROMISE

An unauthorized user may attempt to modify or destroy audit or SEM data, thus removing evidence of unauthorized or malicious activity.

This threat is countered by the following security objectives:

- O.INTEGR – addresses this threat by ensuring that no TOE data will be protected from modification.
- O.STORAGE—addresses this threat by ensuring the TOE is able to protect stored audit records and SEM data from unauthorized modification and deletion.
- OE.CONFID—addresses this threat by ensuring the operational environment provides capabilities to protect the confidentiality of data communicated by the TOE components and by administrative users (including authentication data) to the TOE.

7.1.1.2 T.NO_ACCOUNTABILITY

Authorized users of the TOE perform adverse actions on the TOE, or attempt to perform unauthorized actions, which go undetected.

This threat is countered by the following security objectives:

- O.AUDIT—addresses this threat by ensuring the TOE is able to generate audit records of security relevant events.
- O.AUDIT_REVIEW—supports O.AUDIT in addressing the threat by ensuring the TOE provides capabilities for effective review of stored audit records.
- OE.TIME—supports O.AUDIT by ensuring the operational environment is able to provide the TOE with a reliable time source that can be used to generate time stamps for inclusion within generated audit records.

7.1.1.3 T.UNAUTHORIZED_ACCESS

An unauthorized user may gain access to the TOE security functions and data.

This threat is countered by the following security objectives:

- O.I_AND_A—addresses this threat by ensuring all users of the TOE are identified and authenticated prior to gaining further access to the TOE and its services.
- OE.PLATFORM—supports O.I_AND_A by ensuring the operating system underlying each TOE component protects the component and its configuration from unauthorized access.

7.1.1.4 T.UNAUTHORIZED_ACTIVITY

Authorized users perform unauthorized actions on the TOE.

This threat is countered by the following security objectives:

- O.SECURITY_MANAGEMENT—addresses this threat by providing a mechanism that requires authorized users to have appropriate privileges in order to perform actions on the TOE.

7.1.1.5 T.UNDETECTED_THREATS

Events generated by entities in the IT system indicative of misuse or unauthorized or malicious activity go undetected.

This threat is countered by the following security objectives:

- O.ANALYZER—addresses this threat by ensuring the TOE is able to analyze collected SEM data in order to identify misuse and unauthorized or malicious activity in the IT system being monitored, and be able to record the results of its analysis.
- O.RESPONSE—supports O.ANALYZER in addressing this threat by ensuring the TOE is able to respond to identified misuse and unauthorized or malicious activity.
- O.REVIEW—supports O.ANALYZER in addressing this threat by ensuring the TOE provides capabilities for reviewing the results of its analysis of collected SEM data.
- O.SENSORS—supports O.ANALYZER in addressing this threat by ensuring the TOE provides capabilities to collect SEM data from the IT system it is monitoring and to provide that SEM data to the TOE in a form suitable for the TOE to analyze.
- OE.TIME—supports O.ANALYZER by ensuring the operational environment is able to provide the TOE with a reliable time source that can be used to generate time stamps for inclusion within generated SEM analysis results.

7.1.1.6 A.MANAGE

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

This assumption is satisfied by the following security objective:

- OE.PERSONNEL—this objective satisfies the assumption by ensuring those assigned as authorized administrators are properly trained in operating the TOE.

7.1.1.7 A.PLATFORM

The underlying operating system of each TOE component will protect the component and its configuration from unauthorized access.

This assumption is satisfied by the following security objective:

- OE.PLATFORM—this objective satisfies the assumption by ensuring the operating system underlying each TOE component protects the component and its configuration from unauthorized access.

7.1.1.8 A.PROTECT

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

This assumption is satisfied by the following security objective:

- OE.PHYSICAL—this objective satisfies the assumption by ensuring the TOE is protected from physical attack.

7.1.1.8 A.NOEVIL

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

This assumption is satisfied by the following security objective:

- OE.PERSONNEL—this objective satisfies the assumption by ensuring those assigned as authorized administrators are properly trained in operating the TOE.

7.2 Security Functional Requirements Rationale

All security functional requirements identified in this Security Target are fully addressed in this section and each is mapped to the objective it is intended to satisfy. **Table 21** summarizes the correspondence of functional requirements to TOE security objectives.

	O.ANALYZER	O.AUDIT	O.AUDIT_REVIEW	O.I_AND_A	O.INTEGR	O.RESPONSE	O.REVIEW	O.SECURITY_MANAGEMENT	O.SENSOR	O.STORAGE
FAU_GEN.1		X								
FAU_SAR.1		X	X							
FAU_SAR.2			X							
FAU_SAR.3		X	X							
FAU_SEL.1		X								
FAU_STG.2										X
FAU_STG.4										X
FIA_ATD.1				X						

	O.ANALYZER	O.AUDIT	O.AUDIT_REVIEW	O.I_AND_A	O.INTEGR	O.RESPONSE	O.REVIEW	O.SECURITY_MANAGEMENT	O.SENSOR	O.STORAGE
FIA_UAU.2				X						
FIA_UID.2				X						
FMT_MOF.1								X		
FMT_MTD.1								X		
FMT_SMF.1								X		
FMT_SMR.1								X		
FPT_ITT.1					X					
SEM_ANL_EXT.1	X									
SEM_RCT_EXT.1						X				
SEM_RDR_EXT.1							X			
SEM_LDC_EXT.1									X	
SEM_STG_EXT.1(1)										X
SEM_STG_EXT.1(2)										X
SEM_STG_EXT.1(3)										X
SEM_STG_EXT.2(1)										X
SEM_STG_EXT.2(2)										X
SEM_STG_EXT.2(3)										X

Table 21 - Objectives to Requirement Correspondence

7.2.1.1 O.ANALYZER

The TOE shall analyze collected SEM data in order to identify misuse and unauthorized or malicious activity and shall be able to record the results of its analysis.

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

- SEM_ANL_EXT.1—the ST includes SEM_ANL_EXT.1 to specify the capability to perform correlation, assignment classification, process the metadata and event identification on collected SEM data and to record results of that analysis.

7.1.1.2 O.AUDIT

The TOE shall be able to generate audit records of security-relevant events.

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

- FAU_GEN.1—the ST includes FAU_GEN.1 to specify the capability to generate audit records of security relevant events, and to specify the specific events to be audited and the content of generated audit records of those events.
- FAU_SEL.1—the ST supports FAU_GEN.1 by including FAU_SEL.1 to specify the capability to determine the set of auditable events that are to be audited by the TOE.
- FAU_SAR.1, FAU_SAR.3—the ST includes FAU_SAR.1 and FAU_SAR.3 to specify capabilities to review the contents of the stored audit records and to be able to filter records being reviewed.

7.2.1.3 O. AUDIT_REVIEW

The TOE shall provide a means for authorized users to review the audit records generated by the TOE.

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

- FAU_SAR.1—the ST includes FAU_SAR.1 to specify which roles are to be able to read data from stored audit records.
- FAU_SAR.2—the ST supports FAU_SAR.1 by including FAU_SAR.2 to specify that the ability to read data from stored audit records is restricted to only the roles specified in FAU_SAR.1.
- FAU_SAR.3—the ST supports FAU_SAR.1 by including FAU_SAR.3 to specify capabilities for filtering audit records based on date and time the audit event is recorded, the type of audit event, the subject associated with the audit event, and the outcome of the event, which assists the authorized roles in effectively reviewing the audit trail.

7.2.1.4 O.I _AND_A

The TOE shall provide a means for users to be identified and authenticated before gaining access to TOE services.

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

- FIA_UID.2, FIA_UAU.2—the ST includes FIA_UID.2 and FIA_UAU.2 to specify that users must successfully be identified and authenticated before being able to perform any other TSF-mediated actions.
- FIA_ATD.1—the ST supports FIA_UID.2 and FIA_UAU.2 by including FIA_ATD.1 to ensure user identity and authentication data security attributes are associated with individual users.

7.2.1.5 O. INTEGR

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

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

- FTP_ITT.1—the ST includes FTP_ITT.1 to specify that TSF data is protected from modification and disclosure when it is transmitted between separate parts of the TOE.

7.2.1.6 O. RESPONSE

The TOE shall respond to misuse and unauthorized or malicious activity it identifies based on its configuration.

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

- SEM_RCT_EXT.1—the ST includes SEM_RCT_EXT.1 to specify the capability for the TOE to respond to detected misuse, unauthorized or malicious activity by sending an alarm to a configured destination and taking other actions as specified by the TOE's configuration.

7.2.1.7 O.REVIEW

The TOE shall provide capabilities for effective review of stored SEM data.

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

- SEM_RDR_EXT.1—the ST includes SEM_RDR_EXT.1 to specify capabilities for authorized users to review the results generated by the TOE's analysis functions.

7.2.1.8 O.SECURITY_MANAGEMENT

The TOE shall restrict the ability to perform security management functions on the TOE to authorized administrators having appropriate privileges.

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

- FMT_SMF.1, FMT_SMR.1, FMT_MOF.1, FMT_MTD.1—the ST includes these requirements to specify the security management functions to be provided by the TOE (FMT_SMF.1), to specify security management roles and privileges (FMT_SMR.1), and to specify the restrictions on management of security function behavior and TSF data (FMT_MOF.1, FMT_MTD.1).

7.2.1.9 O.SENSOR

The TOE shall collect SEM data from the IT system the TOE is monitoring and to provide that SEM data to the TOE in a form suitable for the TOE to analyze.

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

- SEM_LDC_EXT.1—the ST includes SEM_LDC_EXT.1 to specify the SEM data collected from the targeted IT system resources.

7.2.1.10 O.STORAGE

The TOE shall protect stored audit records and SEM data from unauthorized modification or deletion.

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

- FAU_STG.2—the ST includes FAU_STG.2 to specify that the TOE protects the audit data from unauthorized deletion as well as guarantee the availability of the audit data in the event of storage exhaustion.
- FAU_STG.4—the ST includes FAU_STG.4 to specify the TOE will prevent the loss of audit data and send an alarm if the audit trail is full.
- SEM_STG_EXT.1(1)-the ST includes SEM_STG_EXT.1(1) to specify that the System Monitor, and Platform Manager components shall ensure that 100% of existing SEM data will be maintained in the event of SEM data storage exhaustion.
- SEM_STG_EXT.1(2)-the ST includes SEM_STG_EXT.1(2) to specify that the AI Engine Server shall ensure that a block of Global administrator-configured size of SEM data will be maintained in the event of SEM data storage exhaustion.
- SEM_STG_EXT.1(3)-the ST includes SEM_STG_EXT.1(3) to specify that the Data Indexer shall ensure that the most recent 80% of the SEM data will be maintained in the event of SEM data storage exhaustion.
- SEM_STG_EXT.2(1)—the ST includes SEM_STG_EXT.2(1) to specify that System Monitors, Data Processors and the Platform Manager will ignore SEM Data an send an alarm if the storage capacity has been reached.

- SEM_STG_EXT.2(2)—the ST includes SEM_STG_EXT.2(2) to specify that AI Engine Servers overwrite the oldest stored SEM Data and send alarms if storage capacity is reached.
- SEM_STG_EXT.2(3)—the ST includes SEM_STG_EXT.2(3) to specify that Data Indexer delete the oldest stored SEM Data and send alarms if storage capacity is reached.

7.3 Security Assurance Requirements Rationale

The ST specifies Evaluation Assurance Level 2. EAL2 was chosen because it is based upon good commercial development practices with thorough functional testing. EAL2 provides the developers and users a moderate level of independently assured security in conventional commercial TOEs. The threat of malicious attacks is not greater than low, the security environment provides physical protection, and the TOE itself offers a very limited interface, offering essentially no opportunity for an attacker to subvert the security policies without physical access. Augmentation was chosen to provide the added assurance that is gained by defining flaw remediation and flaw reporting procedures. Therefore, the target assurance level of EAL 2 augmented with ALC_FLR.2 is appropriate for such an environment.

7.4 Requirement Dependency Rationale

The table below identifies the Security Functional Requirements from Part 2 of the CC and their associated dependencies. It also indicates whether the ST explicitly addresses each dependency. Notes are provided for those cases where the dependencies are satisfied by components which are hierarchical to the specified dependency.

Requirement	Dependencies	How Satisfied
FAU_GEN.1	FPT_STM.1	See TimeStamp Note below.
FAU_SAR.1	FAU_GEN.1	FAU_GEN.1
FAU_SAR.2	FAU_SAR.1	FAU_SAR.1
FAU_SAR.3	FAU_SAR.1	FAU_SAR.1
FAU_SEL.1	FAU_GEN.1, FMT_MTD.1	FAU_GEN.1, FMT_MTD.1
FAU_STG.2	FAU_GEN.1	FAU_GEN.1
FAU_STG.4	FAU_STG.1	FAU_STG.2 (hierarchical to FAU_STG.1)
FIA_ATD.1	None	n/a
FIA_UAU.2	FIA_UID.1	FIA_UID.2 (hierarchical to FIA_UID.1)
FIA_UID.2	None	n/a
FMT_MOF.1	FMT_SMF.1, FMT_SMR.1	FMT_SMF.1, FMT_SMR.1
FMT_MTD.1	FMT_SMF.1, FMT_SMR.1	FMT_SMF.1, FMT_SMR.1
FMT_SMF.1	None	n/a
FMT_SMR.1	FIA_UID.1	FIA_UID.2 (hierarchical to FIA_UID.1)
FPT_ITT.1	None	n/a
SEM_ANL_EXT.1	SEM_LDC_EXT.1	SEM_LDC_EXT.1
SEM_RCT_EXT.1	SEM_ANL_EXT.1	SEM_ANL_EXT.1
SEM_RDR_EXT.1	SEM_LDC_EXT.1, SEM_ANL_EXT.1	SEM_LDC_EXT.1, SEM_ANL_EXT.1
SEM_LDC_EXT.1	None	n/a
SEM_STG_EXT.1	SEM_LDC_EXT.1, SEM_ANL_EXT.1	SEM_LDC_EXT.1, SEM_ANL_EXT.1
SEM_STG_EXT.2	SEM_STG_EXT.1	SEM_STG_EXT.1

Table 22 – TOE SFR Dependency Rationale

TimeStamp Note: The TOE is not a physical device and operates as an application within a process provided by the environment. Thus, the environment is providing resources for the TOE. The environmental objective OE.TIME requires that the TOE's environment provide a reliable timestamp which the TOE can use as needed (e.g., within audit records). Therefore, the functionality specified in the dependency of FAU_GEN.1 upon FPT_STM.1 is available to the TOE from its environment.

7.5 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. **Table 23 - Mapping of TOE SFRs to Security Functions** identifies the relationship between security requirements and security functions.

	Security Audit	Identification and Authentication	Security Event Manager Component Requirements	Security Management	Protection of the TSF
FAU_GEN.1	**X**				
FAU_SAR.1	**X**				
FAU_SAR.2	**X**				
FAU_SAR.3	**X**				
FAU_SEL.1	**X**				
FAU_STG.2	**X**				
FAU_STG.4	**X**				
FIA_ATD.1		**X**			
FIA_UAU.2		**X**			
FIA_UID.2		**X**			
FMT_MOF.1				**X**	
FMT_MTD.1				**X**	

	Security Audit	Identification and Authentication	Security Event Manager Component Requirements	Security Management	Protection of the TSF
FMT_SMF.1				X	
FMT_SMR.1				X	
FPT_ITT.1					X
SEM_ANL_EXT.1			X		
SEM_RCT_EXT.1			X		
SEM_RDR_EXT.1			X		
SEM_LDC_EXT.1			X		
SEM_STG_EXT.1 (1)			X		
SEM_STG_EXT.1 (2)			X		
SEM_STG_EXT.1 (3)			X		
SEM_STG_EXT.2 (1)			X		
SEM_STG_EXT.2 (2)			X		
SEM_STG_EXT.2 (3)			X		

Table 23 - Mapping of TOE SFRs to Security Functions