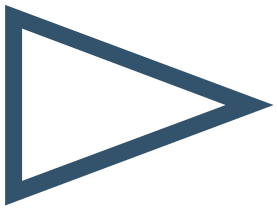


BMC Atrium® Discovery and Dependency Mapping 10.0

Security Target

Version 0.12

10 February 2015

bmcsoftware
BUSINESS RUNS ON I.T.™

Document Revision History

Date	Revision	Author	Changes made
15 February 2013	0.01	Catherine Skrbina	Initial Draft
2 March 2013	0.02	Catherine Skrbina	Second Draft
27 March 2013	0.03	Catherine Skrbina	Third Draft
21 October 2013	0.04	Ron Starman	Circulated for initial internal review
27 February 2014	0.05	Ron Starman	Submitted for Registration
24 March 2014	0.06	TM	Addressed evaluator verdicts
29 May 2014	0.07	TM	Updated TOE diagram, addressed evaluator ORs
24 June 2014	0.08	TM	TOE version change
4 September 2014	0.09	TM	Addressed certifier comments
25 September 2014	0.10	TM	Addressed evaluator comments
5 February 2015	0.11	TM	Addressed evaluator comments
10 February 2015	0.12	TM	Updated CAVP numbers

TABLE OF CONTENTS

1 SECURITY TARGET INTRODUCTION

This section presents Security Target (ST) identification information and an overview of the ST for *BMC Atrium Device and Dependency Mapping 10* (hereinafter referred to as *BMC Atrium Discovery or ADDM*).

An ST contains the information technology (IT) security requirements of an identified Target of Evaluation (TOE) and specifies the functional and assurance security measures offered by that TOE to meet stated requirements. An ST principally defines:

- A security problem expressed as a set of assumptions about the security aspects of the environment, a list of threats that the product is intended to counter, and any known rules with which the product must comply (TOE Security Environment section).
- A set of security objectives and a set of security requirements to address the security problem (Security Objectives and IT Security Requirements sections, respectively).

The structure and content of this ST comply with the requirements specified in Annex A Specification of Security Targets of [CCP1] and Section 11 Class ASE: Security Target evaluation of [CCP3].

1.1 Security Target Reference

ST Title:	BMC Atrium Discovery and Dependency Mapping 10 Security Target
ST Version:	Version 0.12
ST Date:	10 February 2015

1.2 TOE Reference

TOE Identification:	BMC Atrium Discovery and Dependency Mapping 10
TOE Developer	BMC Software, Inc.
TOE Type	Application Discovery and Management

1.3 Document References

The following references are used in this ST:

Abbreviation	Document
[ANSI X9.31]	ANSI Standard X9.31. Digital Signatures Using Reversible Public Key Cryptography for the Financial Services industry (rDSA). January 1998
[CC]	Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 4, September 2012, CCMB-2012-09-(001 to 003)
[CCP1]	Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model; CCMB-2012-09-001, Version 3.1, Revision 4, July 2012
[CCP2]	Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Components; CCMB-2012-09-002, Version 3.1, Revision 4, September 2012
[CCP3]	Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Components; CCMB-2012-09-003, Version 3.1, Revision 4, September 2012
[CEM]	Common Methodology for Information Technology Security Evaluation; CCMB-2012-09-004, Version 3.1, Revision 4, September 2012
[FIPS140-2]	FIPS PUB 140-2. Security Requirements for Cryptographic Modules. May 2001
[FIPS180-3]	FIPS PUB 180-3. Secure Hash Standard (SHS). October 2008
[FIPS186-2]	FIPS PUB 186-2. Digital Signature Standard (DSS). January 2000
[FIPS197]	FIPS PUB 197. Advanced Encryption Standard. November 2001
[FIPS198-1]	FIPS PUB 198-1. The Keyed-Hash Message Authentication Code (HMAC). July 2008
[RFC 4253]	Request for Comments: 4253, The Secure Shell (SSH) Transport Layer Protocol. January 2006

1.4 Document Conventions

Section 8.1 in [CCP1] defines the approved set of operations that can be applied to the CC functional and assurance components: *assignment, refinement*, *selection,* and *iteration.* In this ST, these operations are indicated as follows:

1) The assignment operation is used to assign a specific value to an unspecified parameter, such as the length of a password. Showing the value in square brackets [assignment_value] indicates an assignment. In the case when an assignment operation is embedded in a selection operation, the operations will be denoted as follows: *selection value [assignment_value]*.
2) The refinement operation is used to add detail or refine a requirement, and thus further restricts a requirement. Refinement of security requirements is denoted by **bold text** for new text and ~~strikethrough text~~ for deleted text.
3) The selection operation is used to select one or more options provided by the CC in stating a requirement. Selections are denoted by *underlined italicized text.*
4) Iterated security functional requirements will be identified by appending an additional identifier in round brackets next to their original identifier. For example: FMT_MTD.1(1) and FMT_MTD.1(2).

In addition, the following general conventions are also used in this document:

5) Plain *italicized text* is used to introduce the names of TOE components and specific concepts.
6) ***Bold italicized text*** is used for emphasis.
7) `Text in Courier Font` is used to identify file names, directory paths and BMC Atrium Discovery syntax.

1.5 Document Terminology

1.5.1 CC Terminology

In the CC, many terms are defined in Section 4.1 of [CCP1]. The following terms are a subset of those definitions:

Term	Definition
Authentication data	The information used to verify the claimed identity of a user.
Authorized user	A TOE user who may, in accordance with the SFRs, perform an operation.
External entity	A human or IT entity possibly interacting with the TOE from outside of the TOE boundary.
Identity	A representation uniquely identifying entities (e.g. a user, a process or a disk) within the context of the TOE An example of such a representation is a string. For a human user, the representation can be the full or abbreviated name or a (still unique) pseudonym.
Object	A passive entity in the TOE, that contains or receives information, and upon which subjects perform operations.
Operation (on an object)	A specific type of action performed by a subject on an object.
Role	A predefined set of rules establishing the allowed interactions between a user and the TOE.
Security function policy	A set of rules describing specific security behaviour enforced by the TSF and expressible as a set of SFRs.
Security objective	A statement of an intent to counter identified threats and/or satisfy identified organisation security policies and/or assumptions.
Security requirement	A requirement, stated in a standardised language, which is meant to contribute to achieving the security objectives for a TOE.
Subject	An active entity in the TOE that performs operations on objects.
Target of evaluation	A set of software, firmware and/or hardware possibly accompanied by guidance.

Term	Definition
TOE security functionality	The combined functionality of all hardware, software, and firmware of a TOE that must be relied upon for the correct enforcement of the SFRs.
TSF interface	The means by which external entities (or subjects in the TOE but outside of the TSF) supply data to the TSF, receive data from the TSF and invoke services from the TSF.
User	See external entity defined above.

1.5.2 Abbreviations

The following acronyms are used in this ST:

Term	Definition
ADDM	BMC Atrium Discovery and Dependency Mapping
AES	Advanced Encryption Standard
BAI	Business Application Instance
CBC	Cipher-Block Chaining
CC	Common Criteria
CI	Configuration Item
CMDB	Configuration Management Database
CSEC	Communications Security Establishment Canada
DAC	Discretionary Access Control
DHE	Diffie-Hellman Key Exchange
EAL	Evaluation Assurance Level
ECA	Event Condition Action
FIPS	Federal Information Processing Standards
HTTPS	Hypertext Transfer Protocol Secure
IP	Internet Protocol
ITIL	IT Infrastructure Library®
LDAP	Lightweight Directory Access Protocol
OSI	Operating System Instance
OVF	Open Virtualization Format
PP	Protection Profile
SAR	Security Assurance Requirement
SFP	Security Function Policy
SFR	Security Functional Requirement
SHA	Secure Hash Algorithm
SHS	Secure Hash Standard
SI	Software Instance
SNMP	Simple Network Management Protocol

Term	Definition
SSH	Secure Shell
SSL	Secure Sockets Layer
ST	Security Target
TKU	Technology Knowledge Update
TLS	Transport Layer Security
TOE	Target of Evaluation
TPL	The Pattern Language
TSF	TOE Security Functionality
TSFI	TSF Interface
UI	User Interface
VA	Virtual Appliance
VM	Virtual Machine
WMI	Windows Management Instrumentation

1.5.3 ADDM Terminology

The following additional terms are specific to this ST:

Term	Definition
Business Application Instance	A Business Application Instance (BAI) Node is a datastore node which represents a running instance of a known business application running in the environment. A Business Application is a high level concept, such as payroll, website, or email, that typically depends on a variety of software, databases, application servers, and so on to work. Business Applications are generally not discovered out-of-the-box, as the way in which they are structured and can be discovered varies widely. The Collaborative Application Mapping feature enables you to configure BMC Atrium Discovery to dynamically discover BAIs.
Configuration Item (CI)	Configuration items are hardware and software items discovered by ADDM and subject to management. This should not be confused with the Configuration Items required by ALC_CMS.
Datastore	All data used by the BMC Atrium Discovery system is held in an object database (“datastore”). The datastore treats data as a set of objects and the relationships between them.
Discovery Access	A Discovery Access represents a single access to a Discovery Endpoint. When an endpoint is scanned, a Discovery Access node is created which records information such as the start time, end time, the Discovery Run which contains the Discovery Access, and the previous Discovery Access for the same endpoint to make troubleshooting easier.
Discovery Access Node	For every endpoint (IP address) that is scanned, a Discovery Access Node is created, regardless of whether there is any response from that address. If there is no response or an error, this information is stored on the Discovery Access Node.
Discovery Endpoint	The endpoint of a single Discovery access, the IP address of the discovery target.
Discovery Run	A scan of one or more Discovery endpoints, specified as a collection of IP addresses, or ranges of addresses that are scanned as an entity. For each Discovery Run, a node is created that records information such as the user who started the run, the start and end time, and so forth.
Discovery Target	A computer or network device to be scanned by BMC Atrium Discovery.
Event	A change or action that affects the discovery process, such as a software instance that was created or updated. In BMC Atrium Discovery, the Rules Engine (ECA Engine) executes rules in response to events.
Host Node	A Host node is only created once BMC Atrium Discovery has concluded that a unique host exists. Typically this is after a successful login is achieved.

Term	Definition
Logical Host	A hardware or software host that is contained in a virtual machine (software), a collaborating host in a cluster (hardware) or a blade in a blade server (hardware).
Node	An object in the BMC Atrium Discovery datastore that represents an entity in the environment. Nodes have a kind, such as 'Host', and a number of named attributes. Nodes can be connected to other nodes using relationships. Most node kinds have a key that uniquely identifies the entity in the environment.
Pattern	In BMC Atrium Discovery, the Pattern Language (TPL) is used to create and maintain the model. Each pattern in TPL has a corresponding pattern node in the model, which is related to the nodes that the pattern maintains. Patterns are used to extend the functionality of the reasoning engine.
Relationship	The way that objects are associated with each other. Relationships are defined by the roles represented by each object. They are stored in the datastore in the format: `Node:Role:RelationshipLink:Role:Node`.
Role	The responsibility or actions of the relationship between two nodes. A node with a relationship to another node acts in a role in the relationship, which indicates its part of the relationship. For example, in a 'Dependency' relationship, one node has the role 'Dependent' and the other has the role 'DependedUpon'.
Rules	Small fragments of executable code that run in the Rules Engine in BMC Atrium Discovery. Rules are generated from patterns when they are activated. Additional core rules are distributed with BMC Atrium Discovery.
Software Instance	A Software Instance (SI) Node represents an instance of an off-the-shelf software product or equivalent proprietary item of software. It can correspond to one single running process or a group of processes (possibly on multiple hosts). A Software Instance often corresponds to a licensable entity. Where possible, versions of Software Instances are retrieved and stored. They are created, maintained and destroyed by patterns.

1.6 TOE Overview

1.6.1 General

BMC Atrium Discovery and Dependency Mapping (the Target of Evaluation – "TOE") automatically discovers physical and virtual servers, applications, and network devices and correlates the relationships between them; it simplifies asset management, configuration management, data center consolidation, disaster recovery, and change and release management. It enables administrators to easily answer the question "What do I have?" by discovering details about IT infrastructure, including accurate hardware and software inventory information on servers and network devices. It also helps administrators answer the next question, "How is it connected?" by identifying the interdependencies and automatically building application dependency maps that show how the IT infrastructure supports services.

Discovery and dependency mapping is carried out by the TOE using the following built-in features:

- ***Discovery Engine***: The TOE's discovery engine employs multiple discovery techniques to locate hosts and devices on the network as well as identify their corresponding applications. Authorized TOE users can configure scan levels ranging from searches for a single target to discovery of targets across multiple ranges of IP addresses. The discovery process can be automated through the development and use of patterns. TOE administrators configure user accounts to provide them with varying degrees of access to the TOE's functionality depending on their role (which is determined by the user's membership in TOE security groups).
- ***Data Model***: The TOE data model stores different types of related data in separate areas of the model. In its default configuration, the TOE represents data in four ways:
  - as directly discovered data which has not yet undergone any processing beyond simple parsing.
  - as knowledge information which incorporates static information derived from technology knowledge updates. This information includes general information about products such as hardware details, applications and software products including version information, etc. Knowledge information is included in patterns supplied by BMC.
  - as inferred information that the TOE derives ("reasons") from directly discovered data. This information is typically of primary interest to TOE users. This information includes details about discovered hosts, subnets, software instances (SIs), and business application

instances (BAIs). TOE users can control how the TOE inferences information through the development of their own specific patterns (which are run by the reasoning engine described below).

- as provenance information which is a measure of the quality of the dependency mapping resulting from the ease of which the data can be easily verified. Provenance information is meta-information about how the other information described above came to exist. IT is generated as the reasoning engine builds and maintains the organization's data model.

- ***Reasoning Engine***: The TOE's reasoning engine is an event-based engine which orchestrates and drives the population of the data model through the execution of a series of rules (patterns). Patterns drive the instruction of the reasoning engine, and the pattern language is designed to follow an Event-Condition-Action (ECA) architecture. This approach provides TOE users with a flexible rules-based system that can be customized, changed, and added to without requiring a new version of the reasoning engine. Note however, that TOE users are not expected to need to modify these rules in most deployments.
- **Pattern Language**: The TOE uses a pattern language through which the creation and deletion of items in the data model is configured and expended. The purpose of the pattern language is to abstract the complexity of the ECA-based reasoning engine and present a simple interface for TOE users who are expected to be non-programmers. Through the issue of Technology Knowledge Updates, BMC provides packages of patterns to the end user. The pattern language describes how SIs and BAIs are determined by the TOE.

The TOE uses an agent-less approach to build a complete topology of applications and infrastructure, including servers, operating systems, software, network devices, business applications, and dependencies, and updates the topology as often as needed. It discovers operating systems and environments using standard management protocols, such as Secure Shell (SSH), Windows Management Instrumentation (WMI), and Simple Network Management Protocol (SNMP). For more detailed discovery of Windows-based platforms, the TOE employs a Windows proxy installed on a Windows system. This proxy is required because the methods that are used to access Windows hosts are available only from Windows systems.

The TOE ships as a fully-installed and ready-to-run virtual machine (VM), including a self-contained data store. The TOE is configured simply with an IP range, a schedule, and a set of credentials. The TOE also allows control over the scan level, ranging from a simple sweep scan to full discovery.

As an appliance-based tool which automates discovery of business applications, the TOE is capable of mapping discovered applications onto the underlying physical and virtual IT infrastructure, and determining the critical dependencies between them. The TOE's model-driven, data center indexing techniques cut across previously disparate silos of configuration information, automatically populating and maintaining a data store of the discovered state of Configuration Items (CI) and dependency information.

The TOE automates many system administrator and application management team tasks and also stores customer-sensitive data. At its core, the TOE ensures the confidentiality and integrity of the discovery processes as well as the indexed data itself. The TOE also includes a credential vault which securely stores the credentials used by the TOE to access discovery targets.

Key capabilities of the TOE include:

- Asset and Software License Management
- Configuration Management Database (CMDB)
- Change Impact Analysis
- Data Center Consolidation and Migration
- Disaster Recovery
- Mainframe Cost Optimization
- Service Desk Optimization
- Service Impact Management

1.6.2 TOE Type

BMC Atrium Discovery is an Application Discovery and Management platform.

1.6.3 Required non-TOE Hardware and Software

BMC Atrium Discovery is a software-only TOE that is provided to customers as a virtual appliance (VA) in the Open Virtualization Format (OVF). It relies on the additional non-TOE hardware and software identified in this section to function as specified herein.

The hardware requirements for any given environment depend on the size and amount of activity expected. This section describes minimum and recommended requirements. In most cases, BMC recommends that an analysis of the organization's needs be performed to determine the hardware requirements for the installation.

For complete information about hardware that is compatible with the TOE, refer to the *BMC Atrium Discovery 10.0* document available at http://discovery.bmc.com. BMC Software recommends that customers check the websites of the suppliers of the platforms and supporting components in use at their site to verify that they are still supported. Platforms that are no longer supported by the vendor are not supported by BMC Software.

1.6.3.1 TOE Appliance Requirements

Table 1 lists the minimum hardware host requirements for the TOE.

Table 1. Minimum Host Platform Requirements for the TOE Appliance

Component	Specification
CPU	2 x Intel Xeon E5620 2.40 GHz
Physical Memory	24GB
VM Software	VMware Virtual Infrastructure (ESX/ESXi) – 4.1 and later
Additional Required Software	VMware Tools
Networking	Bridged mode

There are three "classes" of appliance deployment which broadly follow how the TOE is deployed in the field. These classes are differentiated by the number of Operating System Instances (OSIs) that will be scanned by the TOE as follows:

- **Baseline** – A typical baseline as offered by BMC, and will support a maximum of 500 OSIs.
- **Data Center** – A typical large scale deployment, and will support a maximum of 5000 OSIs.
- **Consolidated Enterprise** – An enterprise scale deployment which typically has a Consolidation Appliance taking feeds from many Scanning Appliances. Will typically support a maximum of 20000 OSIs for scanning or consolidation. This deployment may need to adopt a weekly scanning or focused scanning strategy.

Table 2 describes the additional host platform requirements to support the three appliance deployment classes.

Table 2. Additional Requirements for the TOE Appliance

Resource	Baseline	Data Center	Consolidated Enterprise
CPUs	2	4	4 to 8
Physical Memory (GB)	4 to 8	8 to 16	16 to 32
Swap Space (GB)	8 to 16	16 to 32	16 to 32
DB Disk Space (GB) – No backup	100	200	200 to 660
DB Disk Space (GB) – With local backup	200	400	450 to 1300

Note: The full use of a logical CPU (core) is assumed. For example, if eight CPUs are required, then they can be provided in the following ways: a) Eight virtual CPUs in the virtualization platform, such as VMware Infrastructure; b) Four dual core physical CPUs; or c) Two quad core physical CPUs.

1.6.3.2 Windows Proxy Hardware and Software Requirements

The process used by the TOE for discovery of Windows-based hosts follows the model of communication where one device (the master) has unidirectional control over one or more other devices (known as proxies). This unidirectional master proxies control relationship is required because the TOE cannot discover the quality of information required from a Windows system, so it requires that a Windows system perform those tasks itself. As such, the TOE includes a second software component (Windows Proxy) that is used to access Windows-based hosts. The Windows Proxy component is made up of two proxies: an Active Directory Windows proxy that uses a domain account for access, and a Credential Windows proxy that uses specified credentials for access. This software, which is a TOE component, requires that additional software and hardware be present in the IT operational environment.

Table 3 identifies the minimum hardware requirements and the operating system used for Windows Proxy in the evaluated configuration.

Table 3. Host Platform Requirements for the Windows Proxy

Component	Specification
CPU	2GHz Intel Pentium® 4 CPU 512k Cache (or equivalent/better from Intel or other manufacturer)
Physical Memory	At least 2GB
Available Disk Space	At least 60GB
Operating System	Windows 2008 R2

1.6.3.3 Network Time Protocol Service

The TOE requires that the host appliance be configured to synchronize time with one or more NTP services. If significant clock skews occur then it can impact the functioning of the system or even prevent the TOE from starting.

1.6.3.4 TOE Console

The TOE administrators require a console with network access to the TOE through both a web browser and a SSH connection from a terminal or terminal emulator (if using MS-Windows). For Windows-based workstations, the following tools are recommended to be installed on the host console:

- PuTTy (or an equivalent tool) to support SSH connections to the TOE; and
- WinSCP (or an equivalent tool) to secure the transfer of data between the management console and the TOE.

Authorized TOE users can also access the TOE via a web browser to use the information collected by it.

1.7 TOE Description

This section provides context for the TOE evaluation by identifying the product type and describing the evaluated configuration.

1.7.1 Product Type and Evaluated Component Names

The BMC Atrium Discovery system is an Application Discovery and Management platform.

Table 4 identifies the TOE's components included in the evaluated configuration. The “abbreviated name” is used in this Security Target for discussion purposes.

Table 4. BMC Atrium Discovery TOE Component Names and Versions

TOE component name and version	Version	Abbreviated name
BMC Atrium Discovery and Dependency Mapping Appliance	10.0.0.2 build 365935 with OS update 6.14.11.21	*ADDM Appliance, Appliance*
BMC Atrium Discovery and Dependency Mapping Windows Proxy	10.0.0.2 build 365935 with OS update 6.14.11.21	*ADDM Windows Proxy, Windows Proxy*

1.7.1.1 User Guidance

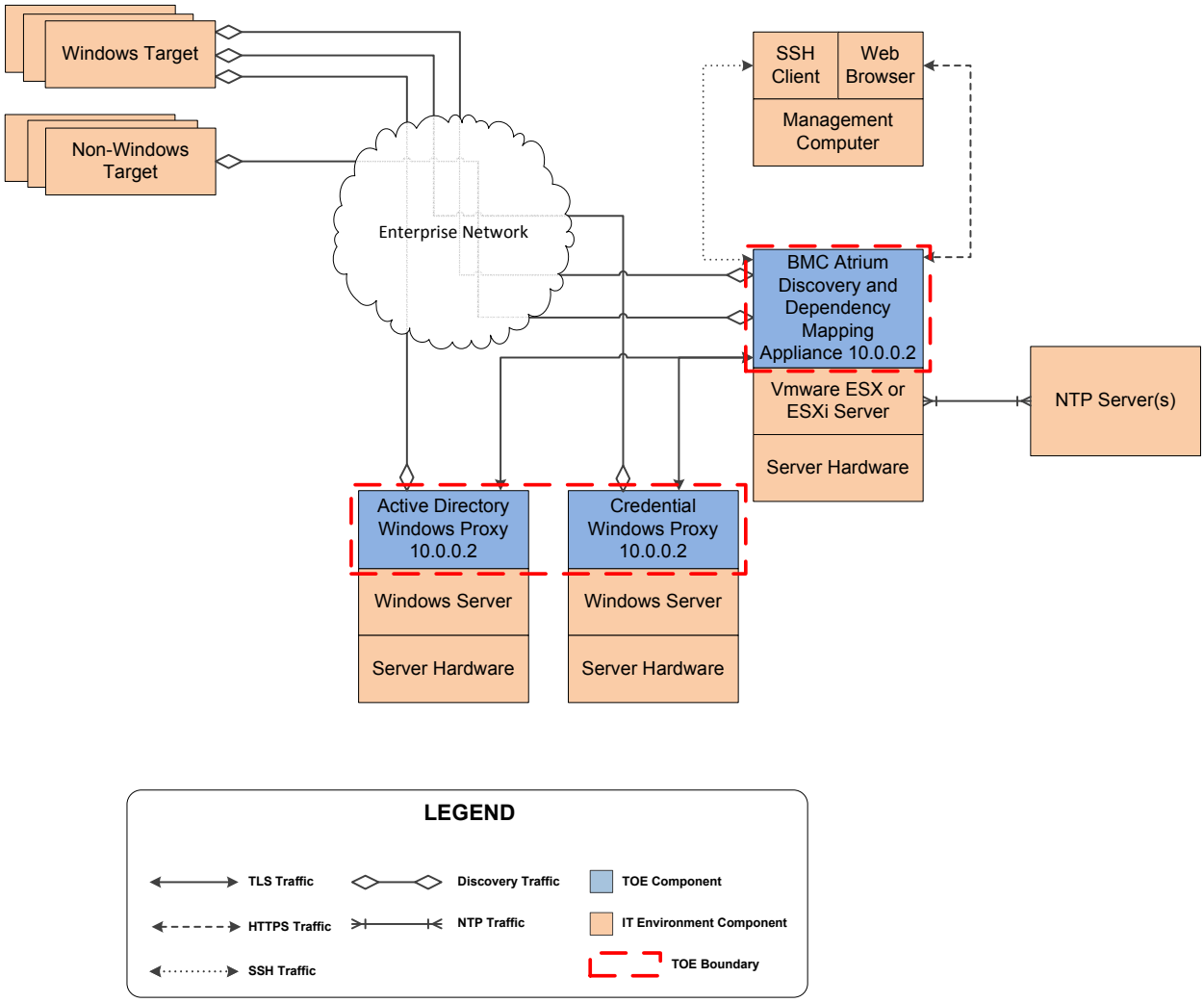
The following guidance is included with the TOE:

- BMC Atrium Discovery Release Notes
- Getting Started with BMC Atrium Discovery
- BMC Atrium Discovery User Guide
- BMC Atrium Discovery Configuration Guide
- BMC Atrium Discovery Deployment Guide
- BMC Atrium Application Mapping Guide
- Security in BMC Atrium Discovery
- Troubleshooting

The latest versions are available at http://discovery.bmc.com/confluence/display/100/Documentation.Physical Scope and Boundary

Figure 1 identifies the physical scope and the physical boundary of the TOE, as well as showing how all the components of the TOE tie together with IT systems in the enterprise.

Figure 1. BMC Atrium Discovery TOE Boundary



1.7.1.2 ADDM Appliance

The ADDM Appliance can be a critical part of an enterprise's data center operations strategy for supporting the availability and maintainability of its IT assets. The ADDM Appliance's core function is to provide asset and application discovery and dependency mapping services externally to IT components used to manage the enterprise's network. The ADDM Appliance automatically discovers physical and virtual servers, applications, and network devices, it determines the relationships between them, and assigns a data quality score to each determined relationship. This approach can significantly reduce IT administrative and management costs for the enterprise while at the same time dramatically simplifying asset management, configuration management, data center consolidation, disaster recovery, and change and release management initiatives.

The ADDM Appliance identifies the interdependencies between IT hosts by analyzing the network communication between them. It then automatically groups them together according to the applications they support. By drilling into an Automatic Host Group, the administrator will see all the servers in the group, how they are communicating, and with whom they are communicating. The administrator can also organize hosts and other discovered items into manual groups specific to the organization's need; for example, to specify a group of servers for a data center migration project.

Using the ADDM Appliance, data center administrators can map the organization's applications showing precisely how IT components (i.e., switches, routers, servers, hardware, and software) correlate to them. Change and problem management processes will also be more efficient since administrators can rely on an automated view of dependencies between all hardware and software components and can quickly determine their impact on the organization's business applications.

1.7.1.3 ADDM Windows Proxy (Windows Proxy)

Because the methods that are used to access Windows hosts are only available from Windows systems, Windows discovery requires a Windows proxy host. Windows discovery is handled in one of the following ways:

- **Credential Windows proxy** – a TOE component that runs on a customer-provided Windows host and uses credentials supplied by the ADDM Appliance to perform Windows discovery.
- **Active Directory Windows proxy** – a TOE component that runs on a customer-provided Windows host that is part of an Active Directory domain or Workgroup. The user that the discovery service runs as, is configured after the Windows proxy is installed. Where that user is configured on hosts in the domain, the Windows proxy can log in and run discovery commands. The Active Directory Windows proxy does not use any credentials entered using the ADDM Appliance User Interface.

The Windows proxy scans Windows hosts on behalf of the discovery service on the ADDM Appliance.

1.7.2 Logical Scope and Boundary

The TOE provides the following security functions:

- Security Audit Data Generation
- Cryptographic Support
- User Data Protection
- Identification and Authentication
- Security Management
- Protection of the TSF
- Trusted Path/Channel
- Discovery and Dependency Mapping

1.7.2.1 Security Audit Data Generation

The TOE generates audit records that identify the date and time of security events, the type of event, subject identity, and the outcome (success or failure as provided in an event summary description) of the event. TOE security audit records are stored in the TOE's internal datastore in groups that are presentable to authorized users on the Audit page of the TOE user interface. The security audit record log is protected from modification or deletion by TOE users by a set of audit permissions that are defined by the TOE administrator. Audit records may be reviewed and sorted by authorized administrators.

1.7.2.2 Cryptographic Support

The TOE includes two FIPS-validated cryptographic modules that provide cryptographic support to the TSF. The modules are used to generate and zeroize keys as well as carry out the necessary encryption, decryption, signing, verifying, and hashing operations needed by TSF's supported communications protocols.

The TOE supports secure communication both within its own scope of control (between TOE components) as well as with other systems and trusted IT entities via the Transport Layer Security (TLSv1) protocol (see Figure 1). This protocol provides protection against unauthorized disclosure and modification via cryptographic mechanisms as described below:

- TLS is used to secure intra-TOE communications between the ADDM Appliance TOE component and Windows Proxy TOE components.
- TLS is also used to secure communications between the ADDM Appliance TOE component and scanning ADDM Appliances within the enterprise network when operating in the consolidated enterprise configuration.

The TOE enables the establishment of a trusted path between the ADDM Appliance TOE component and Administrator management consoles or User consoles located outside the TOE boundary via the Secure Shell (SSHv2.0) and HTTPS protocols (see Figure 1). Communication via these protocols provides protection against unauthorized disclosure and modification via cryptographic mechanisms and the implementation provides assured identification of its end points. The protocols are used as follows:

- SSH is used to establish a secure channel for TOE Administrator Command Line Interface (CLI) access to the ADDM Appliance TOE component from the management console
- HTTPS is used to secure the TOE UI communication channel between the management console and the ADDM Appliance TOE component when used by TOE Administrators and TOE Users

1.7.2.3 User Data Protection

Access to TOE interfaces and data are controlled by the BMC Atrium Discovery Access Control SFP based on the user's membership in a security group.

1.7.2.4 Identification and Authentication

The TOE requires users to be identified and authenticated before completing any security management related actions. Passwords must meet a minimum standard. Once the user is authenticated, the TOE enforces role-based rules and only an authorized Administrator can make changes. In the evaluated configuration, the TOE maintains information about users and their credentials itself. The TOE detects unsuccessful authentication attempts, and when the defined number of unsuccessful authentication attempts has been met, the TOE blocks users from authenticating.

1.7.2.5 Security Management

The TOE provides security management functions that can only be accessed by authorized TOE administrators. The TOE restricts the ability to determine the behavior of, disable, enable, and modify the behavior of the functions (i.e., security policy rules and privileges) by roles (user membership in Security Groups). The TOE also provides the functions necessary for effective management of the TSF. All authorized administrators must login with unique name and password. Access to the management functions is based on the assigned roles (Security Groups).

The TOE enforces an access control Security Functional Policy (SFP) that determines who can access TOE functions and data. This access control SFP is effected through the implementation of Security Groups by the TSF. All TOE users must be a member of one or more Security Groups. Membership in Security Groups defines the TOE functionality that a user is entitled to access. Each TOE user's view of the user interface and the options that the user can access are dependent on the access privileges provided by the SFP as defined by the Security Groups that the user is a member of.

1.7.2.6 Protection of the TSF

Data transferred between different parts of the TOE, or between different instances of the TOE are protected from disclosure while in transit.

1.7.2.7 Trusted Path/Channel

The ADDM Appliance Subsystem secures the browser-based GUI communications between remote users in the IT environment and the ADDM Appliance TOE component using HTTP over TLSv1.0.

1.7.2.8 Discovery and Dependency Mapping

The TOE explores IT systems to identify hardware and software and then creates configuration items (CIs) and determines the relationships between the CIs based on the discovered data. This correlated information is then available for use by other network management components within the organization such as configuration management databases (CMDBs), etc.

1.7.3 Functionalities Excluded from the Evaluated TOE

The following are excluded from this certification:

- The Web Browser on the management console used for administration of the TOE.
- The SSH Client on the management console used for administration of the TOE via the command line interface (CLI).
- Discovery Targets
- Configuration Management Database
- NTP server optionally used to synchronize the ADDM appliance clock
- LDAP server optionally used for external user authentication
- Windows Proxy Manager

2 CONFORMANCE CLAIMS

2.1 Common Criteria Conformance Claim

This Security Target and the TOE it describes are conformant to version 3.1 of Common Criteria for Information Technology Security Evaluation:

- Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model; CCMB-2012-09-001, Version 3.1, Revision 4, September 2012 [CCP1]
- Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Components; CCMB-2012-09-002, Version 3.1, Revision 4, September 2012 [CCP2]
  - Part 2 Extended
- Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Requirements CCMB-2012-09-003, Version 3.1, Revision 4, September 2012 [CCP3]
  - Part 3 Conformant at the Evaluation Assurance Level 2, augmented with ALC_FLR.2 – Flaw reporting procedures.

The Common Methodology for Information Technology Security Evaluation, Version 3.1, Revision 4, September 2012 [CEM] has been taken into account.

2.2 Protection Profile Claim

This Security Target does not claim conformance to a validated Protection Profile.

2.3 Assurance Package Claim

This Security Target and the TOE it describes is conformant to Evaluation Assurance Level 2+ augmented with ALC_FLR.2.

3 SECURITY PROBLEM DEFINITION

3.1 Threats

Table 5 lists threats to the resources to be protected by the TOE. The threat agents to the TOE are considered to be unauthorized users with public knowledge of how the TOE operates and who possess the skills and resources to alter TOE configuration settings, or parameters, or both. The threat agents do not have physical access to the TOE. Mitigation of the threats is through the objectives identified in Section 4.1 Security Objectives.

Table 5. Threats

Threat	Description
T.ACCOUNT	An authorized user of the TOE could gain access to resources or information protected by the TOE, or perform operations for which no access rights have been granted, via user error, system error, or other actions without being detected.
T.EAVESDROPPING	Malicious users could monitor (e.g., Sniff) network traffic in an unauthorized manner.
T.EXCEED_PRIV	Human users of the TOE might attempt to view, modify, or delete TOE objects, or execute or modify applications for which they do not have the prescribed authority, as specified by local policy, in order to disrupt, or otherwise hinder, business operations.
T.MANAGE	Administrators of the TOE might not have utilities sufficient to effectively manage the security functions of the TOE, as specified by local security policy.
T.UNAUTH_ACCESS	An unauthorized user or subject might gain access to TSF data to view, modify, or delete that data, or execute system applications or modify system applications in order to disrupt, or otherwise hinder, business operations.
T.UNKNOWN	Administrators of remote servers might have incomplete information about their IT inventory which leads to inaccurate inventories, failed configuration changes, longer problem resolution times, and ineffective disaster recovery plans.

3.2 Organizational Security Policies

There are no Organizational Security Policies that apply to the TOE.

3.3 Assumptions

The assumptions delineated in Table 6 are required to ensure the security of the TOE:

Table 6. Assumptions

Assumption	Description	Aspect
A.ADMIN	One or more authorized administrators will be assigned to install, configure and manage the TOE and the security of the information it contains.	Personnel
A.NOEVIL	Users of the TOE are not careless, willfully negligent, or hostile and will follow and abide by the instructions provided by the guidance documentation.	Personnel
A.LOCATE	The TOE will be installed in a manner that will limit physical access to authorized users.	Connectivity

Assumption	Description	Aspect
A.LOCK_DOWN	All supporting operational environment components have had all current security patches (if applicable) applied, and the Administrator has configured the inherent component security mechanisms to their most restrictive settings that will still permit TOE functionality and interoperability. Any such patch must not interfere with the correct functioning of TOE's interfaces to the supporting operational environment components.	Connectivity
A.PHYSICAL_PROTECT	The processing resources of the TOE will be located within facilities providing controlled access to prevent unauthorized physical access.	Physical
A.TIME	The operational environment will provide reliable system time to the TOE.	Connectivity

4 SECURITY OBJECTIVES

The purpose of the security objectives is to address all of the security concerns, and to show which security concerns are addressed by the TOE, and which are addressed by the environment. Threats can be directed against the TOE or the security environment or both. Therefore, the CC identifies two categories of security objectives:

- Security objectives for the TOE
- Security objectives for the environment

4.1 Security Objectives for the TOE

This section identifies and describes the security objectives for the TOE, as shown in Table 7.

Table 7. Security Objectives for the TOE

Security Objective	Description
O.ACCOUNTABLE	The TSF must ensure that requests to invoke controlled interfaces are audited so that those users can be held accountable for their actions.
O.AUTHORIZATION	The TSF must ensure that only authorized users and applications gain access to the TOE and its resources.
O.DISCOVERY	The TOE will provide accurate hardware and software inventory information on servers and network devices. The TOE will also identify interdependencies and will provide application dependency maps.
O.DISCRETIONARY_ACCESS	The TSF must limit access to named objects maintained by the TOE to users or applications with authorization and appropriate privileges. The TSF must allow authorized users to specify which users can access their objects and the actions performed on the objects.
O.EAVESDROPPING	The TOE will encrypt TSF data that traverses the network to prevent malicious users from gaining unauthorized access to TOE data.
O.MANAGE	The TSF must provide all of the functions and facilities necessary to support the Authorized Administrators that are responsible for the management of TOE security.

4.2 Security Objectives for the Environment

This section identifies and describes the security objectives for the environment, as shown in Table 8.

Table 8. Security Objectives for the Environment

Objective	Description
OE.ADMIN	One or more authorized administrators will be assigned to install, configure and manage the TOE and the security of the information it contains.
OE.LOCATE	The TOE will be located within controlled access facilities that will prevent unauthorized physical access.
OE.LOCK_DOWN	Those responsible for the TOE must ensure that the all associated supporting components in the operational environment have had all current patches applied, and are configured in the most restrictive way that will still allow TOE access to all supporting operational environment components. They must also assure that any future security patches do not interfere with the correct functioning of TOE's interface to the supporting operational environment components.
OE.NOEVIL	All users of the TOE are not careless, willfully negligent, or hostile and will follow and abide by the instructions provided by the guidance documentation.

Objective	Description
OE.PHYSICAL	Those responsible for the TOE must ensure that the host computer system(s) containing the ADDM Appliance and the Windows Proxy TOE components are protected from physical attack.
OE.TIME	The operational environment must provide correct (i.e., reliable and accurate) system time to the TOE.

4.3 Security Objectives Rationale

This section demonstrates that all security objectives for the TOE are traced back to aspects of the identified threats to be countered by the TOE, and that all objectives for the environment are traced back to assumptions for the environments.

Table 9. Security Objective to Threats and Assumptions Correspondence

	Threats						Assumptions					
	T.ACCOUNT	T.EAVESDROPPING	T.EXCEED_PRIV	T.MANAGE	T.UNAUTH_ACCESS	T.UNKNOWN	A.TIME	A.ADMIN	A.NOEVIL	A.LOCATE	A.LOCK_DOWN	A.PHYSICAL_PROTECT
O.ACCOUNTABLE	X											
O.AUTHORIZATION					X							
O.DISCOVERY						X						
O.DISCRETIONARY_ACCESS			X									
O.EAVESDROPPING		X			X							
O.MANAGE			X	X	X							
OE.ADMIN								X				
OE.LOCATE										X		
OE.LOCK_DOWN											X	
OE.NOEVIL									X			
OE.PHYSICAL												X
OE.TIME							X					

Table 10. Security Objectives Rationale for the TOE

Objective	Threat	Rationale
O.ACCOUNTABLE	**T.ACCOUNT**	O.ACCOUNTABLE mitigates the T.ACCOUNT threat by ensuring that security relevant changes to configuration are documented and are attributable in a non-reputable fashion to the entity responsible for the modification.
O.AUTHORIZATION	**T.UNAUTH_ACCESS**	O.AUTHORIZATION helps to mitigate the threat T.UNAUTH_ACCESS by requiring the TOE to allow only authorized users and applications access to the TOE.
O.DISCOVERY	**T.UNKNOWN**	O.DISCOVERY mitigates T.UNKNOWN by ensuring that Administrators have accurate and up-to-date hardware and software inventory information on the servers and networks devices residing in their IT infrastructure. O.DISCOVERY will also identify interdependencies and provide application dependency maps for the IT infrastructure.

Objective	Threat	Rationale
O.DISCRETIONARY_ACCESS	**T.EXCEED_PRIV**	O.DISCRETIONARY_ACCESS helps to mitigate the threat T.EXCEED_PRIV by using group-based permissions as a mechanism to limit access to TOE objects or applications.
O.EAVESDROPPING	**T.EAVESDROPPING**	O. EAVESDROPPING mitigates T.EAVESDROPPING by ensuring that all communication to and from the TOE or between components of the TOE are not sent unless they are encrypted.
	T.UNAUTH_ACCESS	O.EAVESDROPPING helps to mitigate the threat T.UNAUTH_ACCESS by requiring that all TSF data flowing between the TOE components is encrypted.
O.MANAGE	**T.EXCEED_PRIV**	O.MANAGE mitigates the threat T.EXCEED_PRIV by requiring the TOE to provide the administrative functionality to manage the TOE to prevent unauthorized access.
	T.MANAGE	O.MANAGE mitigates the threat T.MANAGE by providing all of the functions and facilities necessary to support authorized administrators responsible for management of TOE security.
	T.UNAUTH_ACCESS	O.MANAGE mitigates the threat T.UNAUTH_ACCESS by requiring the TOE to provide the administrative functionality to manage the TOE to prevent unauthorized access.

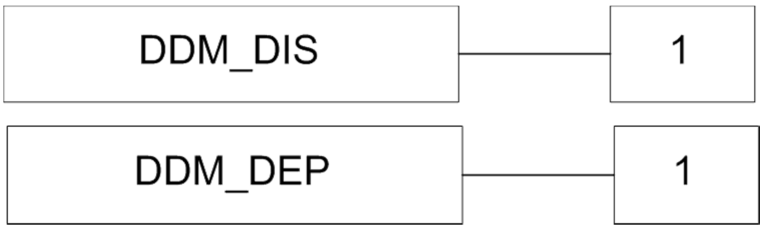
Table 11. Environment Security Objectives Rationale for the TOE

Objective	Assumption	Rationale
OE.ADMIN	**A.ADMIN**	OE.ADMIN maps to A.ADMIN in order to ensure that authorized administrators install, manage and operate the TOE in a manner that maintains its security objectives.
OE.LOCATE	**A.LOCATE**	OE.LOCATE directly maps to A.LOCATE to ensure that those responsible for the TOE locate the TOE in a controlled access facility that will prevent unauthorized physical access.
OE.LOCK_DOWN	**A.LOCK_DOWN**	OE.LOCKED_DOWN meets the assumption A.LOCKED_DOWN. This environmental objective ensures that all supporting operational environment components have had all current security patches applied, and that the authorized administrator has configured the supporting operational component(s) security mechanism(s) to their most restrictive settings that will still permit TOE functionality and interoperability. It also requires the administrator to ensure that any such patch does not interfere with the correct functioning of the TOE's interface to the supporting operational component(s).
OE.NOEVIL	**A.NOEVIL**	OE.NOEVIL directly maps to A.NOEVIL and ensures that all users of the TOE are properly trained in the configuration and usage of the TOE and will follow the guidance provided.
OE.PHYSICAL	**A.PHYSICAL_PROTECT**	OE.PHYSICAL meets the environmental assumption A.PHYSICAL_PROTECT, by requiring that the TOE be located within facilities providing controlled access, to prevent unauthorized physical access.
OE.TIME	**A.TIME**	The environment objective OE.TIME meets the assumption A.TIME, by requiring that the operational environment of the TOE provide reliable system time to the TOE.

5 EXTENDED COMPONENTS DEFINITION

This section specifies the extended SFRs for the TOE. A new class (Discovery and dependency mapping – DDM) was created because the CC does not provide the means to specify the capability for a TOE to discover IT hardware and software, create instances of configuration items and define relationships amongst the discovered data. The DDM class is comprised of two families (discovery – DIS, and determine dependency relationships – DEP) as shown at Figure 2.

Figure 2. DDM: Discovery and dependency mapping class decomposition



5.1 Discovery (DDM_DIS)

Family Behavior: This family defines requirements for ensuring that the TOE performs discovery of IT hardware and software, including applications, on the network. This family includes one component DDM_DIS.1.

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

- Start and stop discovery scans
- Effect changes to discovery configuration settings for events, scan levels, and patterns.

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

- Minimal: discovery start and stop, adding and removing discovery events, changes to discovery scan levels, and modification (including addition of new and removal) of discovery patterns.

DDM_DIS.1	**Discovery**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
DDM_DIS.1.1	The TSF shall be able to [selection: *manually*, *automatically*, [assignment: *other modes of operation*]] discover [selection: *physical servers*, *virtual servers*, *applications*, *network devices*, [assignment: *other network components*] that are present on the network.

5.2 Determine Dependency Relationships (DDM_DEP)

Family Behavior: This family defines requirements for ensuring that the TOE determines the dependency relationships that exist between discovered IT hardware and software, including applications, on the network. This family includes only one component DDM_DEP.1.

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

- There are no management activities foreseen.

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

- There are no audit events foreseen.

DDM_DEP.1	Determine dependency relationships
Hierarchical to:	No other components.
Dependencies:	DDM_DIS.1 Discovery
DDM_DIS.1.1	The TSF shall be able to create configuration items (CIs) and determine relationships that exist between CIs from the data discovered by the TOE.

6 SECURITY REQUIREMENTS

6.1 Security Functional Requirements

Table 12 identifies the Security Functional Requirements (SFR) claimed by the TOE. The table has been augmented to summarize the operations performed on each SFR selected for the TOE. The operations are identified as follows: A = Assignment, S = Selection, R = Refinement and I = Iteration.

Table 12. TOE Security Functional Requirements

Class	Functional component	A	S	R	I
Security Audit (FAU)	**FAU_GEN.1** Audit data generation	X	X	X	
	FAU_GEN.2 User identity association				
	FAU_SAR.1 Audit review	X			
	FAU_SAR.2 Restricted audit review				
	FAU_SAR.3 Selected audit review	X		X	
	FAU_STG.2 Guarantees of audit data availability	X	X		
	FAU_STG.3 Action in case of possible audit data loss	X			
Cryptographic Support (FCS)	**FCS_CKM.1(1)** Cryptographic key generation (RSA)	X			X
	FCS_CKM.1(2) Cryptographic key generation (AES)	X			X
	FCS_CKM.4 Cryptographic key destruction	X			
	FCS_COP.1(1) Cryptographic operation (AES)	X			X
	FCS_COP.1(2) Cryptographic operation (RSA)	X			X
	FCS_COP.1(3) Cryptographic operation (SHS)	X		X	X
	FCS_COP.1(4) Cryptographic operation (HMAC)	X		X	X
User Data Protection (FDP)	**FDP_ACC.1** Subset access control	X			
	FDP_ACF.1 Security attribute based access control	X			
Identification and Authentication (FIA)	**FIA_AFL.1** Authentication failure handling	X	X		
	FIA_ATD.1 User attribute definition	X			
	FIA_SOS.1 Verification of secrets	X			
	FIA_UAU.2 User authentication before any action				
	FIA_UAU.7 Protected authentication feedback	X			
	FIA_UID.2 User identification before any action				
Security Management (FMT)	**FMT_MOF.1** Management of security functions behavior	X	X		
	FMT_MSA.1 Management of security attributes	X	X		
	FMT_MSA.3 Static attribute initialisation	X	X		
	FMT_MTD.1 Management of TSF data	X	X		
	FMT_SAE.1 Time-limited authorisation	X		X	
	FMT_SMF.1 Specification of management functions	X			
	FMT_SMR.1 Security roles	X			
Protection of the TSF (FPT)	**FPT_ITC.1** Inter-TSF confidentiality during transmission				

Class	Functional component	A	S	R	I
	FPT_ITT.1 Basic internal TSF data transfer protection		X	X	
Trusted Path/Channels (FTP)	**FTP_TRP.1** Trusted path	X	X		
Discovery and Dependency Mapping (DDM)	**DDM_DIS.1** Discovery		X		
	DDM_DEP.1 Determine dependency relationships				

6.1.1 Security Audit (FAU)

FAU_GEN.1	**Audit data generation**
Hierarchical to:	No other components.
Dependencies:	FPT_STM.1 Reliable time stamps
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) [User login/logout; and d) All user-initiated events that modify the state or behaviour of the TOE].
FAU_GEN.1.2	The TSF shall record within each audit record at least the following information: a) Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure **as described in the audit record Summary description**) 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, [Event group and User group, where applicable].

FAU_GEN.2	**User identity association**
Hierarchical to:	No other components.
Dependencies:	FAU_GEN.1 Audit data generation FIA_UID.1 Timing of identification
FAU_GEN.2.1	For audit events resulting from actions of identified users, the TSF shall be able to associate each auditable event with the identity of the user that caused the event.

FAU_SAR.1	**Audit review**
Hierarchical to:	No other components.
Dependencies:	FAU_GEN.1 Audit data generation
FAU_SAR.1.1	The TSF shall provide [users in the Security Groups: admin, system] 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.

Application Note: Additional security groups may be created with read permission for audit records. Also, users in other security groups may be provided with audit read permissions.

FAU_SAR.2	**Restricted audit review**
Hierarchical to:	No other components.
Dependencies:	FAU_SAR.1 Audit review
FAU_SAR.2.1	The TSF shall prohibit all users read access to the audit records, except those users that have been granted explicit read-access.

FAU_SAR.3	**Selectable audit review**
Hierarchical to:	No other components.
Dependencies:	FAU_SAR.1 Audit review
FAU_SAR.3.1	The TSF shall provide the ability to ~~apply~~ [search and sort] ~~of~~ audit data based on [event, event group, userID, date and time].

FAU_STG.2	**Guarantees of audit data availability**

Hierarchical to:	FAU_STG.1 Protected audit trail storage
Dependencies:	FAU_GEN.1 Audit data generation
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 [the most recently logged] stored audit records will be maintained when the following conditions occur: ***audit storage exhaustion***.

FAU_STG.3	**Action in case of possible audit data loss**
Hierarchical to:	No other components.
Dependencies:	FAU_STG.1 Protected audit trail storage
FAU_STG.3.1	The TSF shall [shutdown the TOE and prevent TOE restart] if the audit trail exceeds [an Administrator set limit].

6.1.2 Cryptographic Support (FCS)

FCS_CKM.1(1)	**Cryptographic key generation (RSA)**
Hierarchical to:	No other components.
Dependencies:	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction
FCS_CKM.1.1(1)	The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [RSA] and specified cryptographic key sizes [2048 bits] that meet the following: [ANSI X9.31].

FCS_CKM.1(2)	**Cryptographic key generation (AES)**
Hierarchical to:	No other components.
Dependencies:	[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation] FCS_CKM.4 Cryptographic key destruction
FCS_CKM.1.1(2)	The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm [RNG] and specified cryptographic key sizes [256 bits] that meet the following: [ANSI X9.31].

FCS_CKM.4	**Cryptographic key destruction**
Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation]
FCS_CKM.4.1	The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method [zeroization] that meets the following: [FIPS140-2].

FCS_COP.1(1)	**Cryptographic operation (AES)**
Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or

	FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction
FCS_COP.1.1(1)	The TSF shall perform [encryption and decryption] in accordance with a specified cryptographic algorithm [AES] and cryptographic key sizes [256 bits] that meet the following: [FIPS197].

FCS_COP.1(2)	**Cryptographic operation (RSA)**
Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction
FCS_COP.1.1(2)	The TSF shall perform [digital signature generation / verification] in accordance with a specified cryptographic algorithm [RSA] and cryptographic key sizes [2048 bits] that meet the following: [FIPS186-2].

FCS_COP.1(3)	**Cryptographic operation (SHS)**
Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction
FCS_COP.1.1(3)	The TSF shall perform [message hashing] in accordance with a specified cryptographic algorithm [SHA-1] and ~~cryptographic key~~ **message digest** sizes [160 bits] that meet the following: [FIPS180-3].

FCS_COP.1(4)	**Cryptographic operation (HMAC)**
Hierarchical to:	No other components.
Dependencies:	[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation] FCS_CKM.4 Cryptographic key destruction
FCS_COP.1.1(4)	The TSF shall perform [keyed-hash message authentication] in accordance with a specified cryptographic algorithm [HMAC-SHA-1]**,** ~~and cryptographic~~ key sizes [160 bits]**, and message digest sizes [160 bits]** that meet the following: [FIPS198-1 and FIPS180-3].

6.1.3 User Data Protection (FDP)

FDP_ACC.1	**Subset access control**
Hierarchical to:	No other components.
Dependencies:	FDP_ACF.1 Security attribute based access control
FDP_ACC.1.1	The TSF shall enforce the [BMC Atrium Discovery Access Control SFP] on [: a) Subjects: all TOE users b) Objects: (1) Browser-based User Interface, (2) Command Line Interface, (2) TOE's datastore and the datastore partitions, (3) Discovery credentials,

	(4) Audit log, (5) CMDB export data, and (6) TOE configuration settings c) Operations: all user actions].

FDP_ACF.1	**Security attribute based access control**
Hierarchical to:	No other components.
Dependencies:	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialisation
FDP_ACF.1.1	The TSF shall enforce the [BMC Atrium Discovery Access Control SFP] to objects based on the following: [Subject: TOE User Subject Attributes: a) user membership in Security Groups, and b) specific additional Security Permissions, if any, provided to individual users by an authorized TOE Administrator Object: (1) Browser-based User Interface, (2) Command Line Interface, (2) TOE's datastore and the datastore partitions, (3) Discovery credentials, (4) Audit log, (5) CMDB export data, and (6) TOE configuration settings Object Attributes: none].
FDP_ACF.1.2	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [a user will be granted access to objects/resources if the user belongs to a Security Group with the required permission, or if the user has been specifically granted the required security permission by an authorized TOE Administrator].
FDP_ACF.1.3	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [none].
FDP_ACF.1.4	The TSF shall explicitly deny access of subjects to objects based on the following additional rules: [none].

6.1.4 Identification and Authentication (FIA)

FIA_AFL.1	**Authentication failure handling**
Hierarchical to:	No other components.
Dependencies:	FIA_UAU.1 Timing of authentication
FIA_AFL.1.1	The TSF shall detect when *an administrator configurable positive integer within* [1 to 5] unsuccessful authentication attempts occur related to [a user attempting to authenticate to the TOE].
FIA_AFL.1.2	When the defined number of unsuccessful authentication attempts has been *met*, the TSF shall [block that user account from authenticating to the TOE for the period of time specified by the Administrator].

FIA_ATD.1	**User attribute definition**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FIA_ATD.1.1	The TSF shall maintain the following list of security attributes belonging to individual users: [user name, password, and role (Security Group membership)].

FIA_SOS.1	Verification of secrets
Hierarchical to:	No other components
Dependencies:	No dependencies
FIA_SOS.1.1	The TSF shall provide a mechanism to verify that secrets meet [the following administrator-specified quality metrics: a) minimum password length; b) minimum password re-use history; and c) password complexity rules.]

FIA_UAU.2	User authentication before any action
Hierarchical to:	FIA_UAU.1 Timing of authentication
Dependencies:	FIA_UID.1 Timing of identification
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.

FIA_UAU.7	Protected authentication feedback
Hierarchical to:	No other components
Dependencies:	FIA_UAU.1 Timing of authentication
FIA_UAU.7.1	The TSF shall provide only [the feedback described below] to the user while the authentication is in progress**:** [a) when authenticating via the browser-based User Interface, feedback to the user shall be obscured; and b) when authenticating via the Command Line Interface, feedback to the user shall be suppressed (no information regarding the length of the password being entered shall be provided)].

FIA_UID.2	User identification before any action
Hierarchical to:	FIA_UID.1 Timing of identification
Dependencies:	No dependencies.
FIA_UID.2.1	The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

6.1.5 Security Management (FMT)

FMT_MOF.1	Management of security functions behaviour
Hierarchical to:	No other components.
Dependencies:	FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions
FMT_MOF.1.1	The TSF shall restrict the ability to *determine the behaviour of, modify the behaviour of* the functions [listed below] to [TOE Administrators]: [a) configure the TOE for operation, b) manage TOE users, c) manage Security Groups, d) manage Security Group Permissions, e) manage TOE security policies, and f) configure TOE user Identification and Authentication mechanisms.]

FMT_MSA.1	Management of security attributes
Hierarchical to:	No other components.
Dependencies:	[FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control] FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions
FMT_MSA.1.1	The TSF shall enforce the [BMC Atrium Discovery Access Control SFP] to restrict the ability to *change_default, query, modify, delete* the security attributes [user membership in Security Groups and assignment of individual Security Permissions to users] to [TOE Administrators].

FMT_MSA.3	Static attribute initialisation
Hierarchical to:	No other components.
Dependencies:	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles
FMT_MSA.3.1	The TSF shall enforce the [BMC Atrium Discovery Access Control SFP] to provide *restrictive* default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2	The TSF shall allow ~~the~~ [users associated with an Administrator role] to specify alternative initial values to override the default values when an object or information is created.

FMT_MTD.1	Management of TSF data
Hierarchical to:	No other components.
Dependencies:	FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions
FMT_MTD.1.1	The TSF shall restrict the ability to *change_default, query, modify, delete, [or export]* the [TSF data listed in Table 13] to [TOE users based on their membership in Security Groups].

Table 13. Management of TSF Data

TSF Data	Change Default	Query	Modify	Delete	Export
Audit logs	✓	✓		✓	✓
Security Groups	✓	✓	✓	✓	
Security Permissions associated with a user	✓	✓	✓	✓	
Discovery Patterns	✓	✓	✓	✓	
TOE configuration settings	✓	✓	✓		
TOE security policy settings	✓		✓		

FMT_SAE.1	Time-limited authorization
Hierarchical to:	No other components.
Dependencies:	FMT_SMR.1 Security roles

	FPT_STM.1 Reliable time stamps
FMT_SAE.1.1	The TSF shall restrict the capability to specify an expiration time for [user account life and passwords] to [authorized Administrators].
FMT_SAE.1.2	For each of these security attributes, the TSF shall ~~be able to~~ [enable authorized Administrators to: a) reactivate expired user accounts; and b) reset expired passwords] after the expiration time for the indicated security attribute has passed.

FMT_SMF.1	**Specification of Management Functions**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FMT_SMF.1.1	The TSF shall be capable of performing the following management functions: [a) Maintain TOE users who have read access rights to the Audit log. b) Add or remove Security Permissions from Security Groups to enforce explicit access or denial-based decisions to TSF objects for users. c) Specify a threshold for unsuccessful authentication attempts by users, and the actions to be taken by the TSF in the event that this threshold is met. This includes setting the lockout time period. d) Manage the Security Groups that TOE users are members of. e) Manage password quality metrics and password expiry periods. f) Manage the creation, deletion (including the specification for automatic expiration), and reactivation of TOE user accounts. g) Manage the configuration of TLS parameters used to secure intra TOE communications. h) Manage the configuration of TLS and SSH parameters used to secure the UI and CLI between the Appliance and the management console. i) Start and stop discovery scans. j) Manage discovery configuration changes. k) Change user passwords. l) Manage audit logs. m) Manage TSF data].

FMT_SMR.1	**Security roles**
Hierarchical to:	No other components.
Dependencies:	FIA_UID.1 Timing of identification
FMT_SMR.1.1	The TSF shall maintain the roles [User and Administrator as defined by each user's membership in one or more Security Groups].
FMT_SMR.1.2	The TSF shall be able to associate users with roles.

Application Note: Association of users with a role per FMT_SMR.1.2 means user membership in one or more Security Groups.

6.1.6 Protection of the TSF (FPT)

FPT_ITC.1	**Inter-TSF confidentiality during transmission**
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FPT_ITC.1.1	The TSF shall protect all TSF data transmitted from the TSF to another trusted IT product from unauthorised disclosure during transmission.

FPT_ITT.1	Basic internal TSF data transfer protection
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FPT_ITT.1.1	The TSF shall protect TSF data from *disclosure* **and** *modification* when it is transmitted between separate parts of the TOE.

6.1.7 Trusted Path/Channels (FTP)

FTP_TRP.1	Trusted path
Hierarchical to:	No other components.
Dependencies:	No dependencies.
FTP_TRP.1.1	The TSF shall provide a communication path between itself and *remote* users that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from *disclosure*.
FTP_TRP.1.2	The TSF shall permit *remote users* to initiate communication via the trusted path.
FTP_TRP.1.3	The TSF shall require the use of the trusted path for *initial user authentication, [and all user interaction with the TSF via either the browser-based UI or the CLI]*.

6.1.8 Discovery and Dependency Mapping (DDM)

DDM_DIS.1	Discovery
Hierarchical to:	No other components.
Dependencies:	No dependencies.
DDM_DIS.1.1	The TSF shall be able to *automatically* discover *physical servers*, *virtual servers*, *applications*, *network devices* that are present on the network.

DDM_DEP.1	Determine dependency relationships
Hierarchical to:	No other components.
Dependencies:	DDM_DIS.1 Discovery
DDM_DIS.1.1	The TSF shall be able to create configuration items (CIs) and determine relationships that exist between CIs from the data discovered by the TOE.

6.2 Security Assurance Requirements

The TOE satisfies the Security Assurance Requirements (SARs) delineated in Table 14.

Table 14. TOE Security Assurance Requirements

Class	Assurance Component
Development (ADV)	**ADV_ARC.1** Security architecture description
	ADV_FSP.2 Security-enforcing functional specification
	ADV_TDS.1 Basic design
Guidance Documents (AGD)	**AGD_OPE.1** Operational user guidance
	AGD_PRE.1 Preparative procedures
Life-Cycle Support (ALC)	**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
Security Target Evaluation (ASE)	**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
Tests (ATE)	**ATE_COV.1** Evidence of coverage
	ATE_FUN.1 Functional testing
	ATE_IND.2 Independent testing – sample
Vulnerability Assessment (AVA)	**AVA_VAN.2** Vulnerability analysis

6.3 Security Requirements Rationale

6.3.1 Security Functional Requirements Rationale

Table 15 maps claimed SFRs to security objectives, and Table 16 provides rationale to justify the mappings.

Table 15. Security Objectives to Requirements Correspondence

	O.ACCOUNTABLE	O.AUTHORIZATION	O.DISCOVERY	O.DISCRETIONARY_ACCESS	O.EAVESDROPPING	O.MANAGE
FAU_GEN.1 Audit data generation	X					
FAU_GEN.2 User identity association	X					
FAU_SAR.1 Audit review	X					
FAU_SAR.2 Restricted audit review	X					
FAU_SAR.3 Selected audit review	X					
FAU_STG.2 Guarantees of audit data availability	X					
FAU_STG.3 Action in case of possible audit data loss	X					
FCS_CKM.1(1) Cryptographic key generation (RSA)					X	
FCS_CKM.1(2) Cryptographic key generation (AES)					X	
FCS_CKM.4 Cryptographic key destruction					X	
FCS_COP.1 (1) Cryptographic operation (AES)					X	
FCS_COP.1 (2) Cryptographic operation (RSA)					X	
FCS_COP.1 (3) Cryptographic operation (SHS)					X	
FCS_COP.1 (4) Cryptographic operation (HMAC)					X	
FDP_ACC.1 Subset access control				X		
FDP_ACF.1 Security attribute based access control				X		
FIA_AFL.1 Authentication failure handling		X				
FIA_ATD.1 User attribute definition		X				
FIA_SOS.1 Verification of secrets		X				
FIA_UAU.2 User authentication before any action		X				
FIA_UAU.7 Protected authentication feedback		X				
FIA_UID.2 User identification before any action		X				
FMT_MOF.1 Management of security functions behavior						X
FMT_MSA.1 Management of security attributes						X
FMT_MSA.3 Static attribute initialization						X
FMT_MTD.1 Management of TSF data						X
FMT_SAE.1 Time-limited authorisation						X
FMT_SMF.1 Specification of management functions						X
FMT_SMR.1 Security roles						X

	O.ACCOUNTABLE	O.AUTHORIZATION	O.DISCOVERY	O.DISCRETIONARY_ACCESS	O.EAVESDROPPING	O.MANAGE
FPT_ITC.1 Inter-TSF confidentiality during transmission					X	
FPT_ITT.1 Basic internal TSF data transfer protection					X	
FTP_TRP.1 Trusted path					X	
DDM_DIS.1 Discovery			X			
DDM_DEP.1 Determine dependency relationships			X			

Table 16. Security Functional Requirements Rationale for the TOE

Objective	SFR	Rationale
O.ACCOUNTABLE	**FAU_GEN.1**	FAU_GEN.1 defines the security relevant events that will be recorded by the TOE along with the details of the event that will be recorded.
	FAU_GEN.2	FAU_GEN.2 requires the TOE to associate each auditable event with the identity of the user that caused the event.
	FAU_SAR.1	FAU_SAR.1 provides the capability to read information from the audit records.
	FAU_SAR.2	FAU_SAR.2 requires that there are no other users except those that have been identified in FAU_SAR.1 Audit review that can read the information.
	FAU_SAR.3	FAU_SAR.3 requires that TOE will apply selection and ordering based on the contents of the audit records.
	FAU_STG.2	FAU_STG.2 requires that the TOE will protect the audit records stored in the audit trail and ensure that audit records are not lost when the log gets full.
	FAU_STG.3	FAU_STG.3 requires that the TSF will shut the TOE down when an Administrator-set threshold has been met to ensure that no audit records are lost.
O.AUTHORIZATION	**FIA_AFL.1**	FIA_AFTL.1 enables the TOE administrator to define an upper bound to the number of times a user may unsuccessfully attempt to authenticate to the TOE before being blocked for an administrator-defined period of time.
	FIA_ATD.1	FIA_ATD.1 specifies the security attributes that should be maintained at the level of the user. This means that the security attributes listed are assigned to and can be changed at the level of the user. In other words, changing a security attribute in this list associated with a user should have no impact on the security attributes of any other user.
	FIA_SOS.1	Secrets contain the authentication data provided by the user for an authentication mechanism that is based on knowledge the user possesses. FIA_SOS.1 is used to ensure that the externally-generated secret adheres to the organization's minimum standard to provide assurance that only an authorized user can gain access to the TOE and its resources.
	FIA_UAU.2	FIA_UAU.2 requires that users be authenticated before being provided any access to the TOE and resources protected by the TSF.
	FIA_UAU.7	FIA_UAU.7 limits the amount of information that is fed back to a user attempting to authenticate to the TOE.
	FIA_UID.2	FIA_UID.2 requires a user be identified before any access to the TOE and resources protected by the TOE is allowed.

Objective	SFR	Rationale
O.DISCOVERY	**DDM_DIS.1**	DDM_DIS.1 automatically discovers hardware and software on the network and generates an inventory of this information.
	DDM_DEP.1	DDM_DEP.1 uses the inventory information generated by DDM_DIS.1 to identify the interdependencies between the discovered configuration items resulting in application dependency maps.
O.DISCRETIONARY_ ACCESS	**FDP_ACC.1**	FDP_ACC.1 requires the TOE to prevent unauthorized access to TOE resources by enforcing the BMC Atrium Discovery Access Control Policy.
	FDP_ACF.1	FDP_ACF.1 requires the TOE to enforce the BMC Atrium Discovery Access Control Policy on the protected TOE resources and requires the authorized administrators to configure user access rights accordingly.
O.EAVESDROPPING	**FCS_CKM.1(1)**	FCS_CKM.1(1) requires key generation for RSA in support of TLS and SSH sessions carried out by the TSF.
	FCS_CKM.1(2)	FCS_CKM.1(2) requires the generation of AES keys for use in TSL and SSH sessions carried out by the TSF.
	FCS_CKM.4	FCS_CKM.4 requires destruction of encryption keys used to manage remote sessions of the TOE.
	FCS_COP.1(1)	FCS_COP.1 (1) requires encryption and decryption with AES in support of TLS and SSH sessions of the TOE.
	FCS_COP.1(2)	FCS_COP.1 (2) requires cryptographic signature services with RSA in support of TLS and SSH sessions of the TOE.
	FCS_COP.1(3)	FCS_COP.1 (3) requires cryptographic hashing services with SHA-1 in support of TLS and SSH sessions of the TOE.
	FCS_COP.1(4)	FCS_COP.1 (4) requires keyed-hash message authentication with HMAC-SHA-1 in support of TLS and SSH sessions of the TOE.
	FPT_ITC.1	FPT_ITC.1 will ensure that the traffic between the ADDM Appliance TOE component and any Scanning BMC Atrium Discovery Appliances in the network are encrypted using TLS.
	FPT_ITT.1	FPT_ITT.1 Ensures communications between components of the TOE (ADDM Appliance and Windows Proxy) are encrypted using TLS.
	FTP_TRP.1	FPT_TRP.1 ensures communications between the TOE and the user are encrypted using HTTP over TLS (HTTPS) for users accessing the TOE via the browser-based UI, or SSH for administrative users accessing the TOE via its CLI.
O.MANAGE	**FMT_MOF.1**	FMT_MOF.1 allows the authorized users (roles) to manage the behavior of functions in the TSF that use rules or have specified conditions that may be manageable.
	FMT_MSA.1	FMT_MSA.1 restricts the assignment of security attributes of users and resources to the authorized administrators.
	FMT_MSA.3	FMT_MSA.3 allows the authorized administrators to override the default values set for security attributes when creating user accounts.
	FMT_MTD.1	FMT_MTD.1 restricts the ability to [perform functions specified in tables 17 and 18] the [TSF data specified in tables 17 and 18] to [default roles specified in tables 17 and 18, and other roles as specified].
	FMT_SAE.1	FMT_SAE.1 enables the TOE Administrator to specify time limits for both user account life and passwords. This contributes to how O.MANAGE mitigates unauthorized users from gaining access to the TOE or TSF data.
	FMT_SMF.1	FMT_SMF.1 requires that the TOE provide the ability to manage its security functions including the management of user accounts and user access rights, TOE resources and security information recorded in the audit logs.
	FMT_SMR.1	FMT_SMR.1 requires the TOE to provide the ability to set roles for security relevant authority as well as to restrict the ability to define and assign roles to authorized administrators.

6.3.2 Rationale for SFR Dependencies

Table 17 is a cross-reference of the functional components, their related dependencies, and whether the dependency was satisfied.

Table 17. SFR Dependency Status

SFR	Dependencies	Fulfilled by SFRs in this ST
FAU_GEN.1	FPT_STM.1	This dependency is fulfilled by OE.TIME in the operational environment.
FAU_GEN.2	FAU_GEN.1	FAU_GEN.1
	FIA_UID.1	FIA_UID.2
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_STG.2	FAU_GEN.1	FAU_GEN.1
FAU_STG.3	FAU_STG.1	FAU_STG.2
FCS_CKM.1(1)	[FCS_CKM.2 or FCS_COP.1]	FCS_COP.1(2)
	FCS_CKM.4	FCS_CKM.4
FCS_CKM.1(2)	[FCS_CKM.2 or FCS_COP.1]	FCS_COP.1(1)
	FCS_CKM.4	FCS_CKM.4
FCS_CKM.4	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	FCS_CKM.1(1), FCS_CKM.1(2)
FCS_COP.1(1)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	FCS_CKM.1(2)
	FCS_CKM.4	FCS_CKM.4
FCS_COP.1(2)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	FCS_CKM.1(1)
	FCS_CKM.4	FCS_CKM.4
FCS_COP.1(3)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	N/A - conducting a hashing operation (creating a message digest) does not require the generation of a key per FCS_CKM.1
	FCS_CKM.4	N/A – conducting a hashing operation does not require the destruction of a cryptographic key per FCS_CKM.4
FCS_COP.1 (4)	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	FCS_CKM.1(2)
	FCS_CKM.4	FCS_CKM.4
FDP_ACC.1	FDP_ACF.1	FDP_ACF.1
FDP_ACF.1	FDP_ACC.1	FDP_ACC.1
	FMT_MSA.3	FMT_MSA.3
FIA_AFL.1	FIA_UAU.1	FIA_UAU.2
FIA_ATD.1	None	N/A
FIA_SOS.1	None	N/A
FIA_UAU.2	FIA_UID.1	FIA_UID.2
FIA_UAU.7	FIA_UAU.1	FIA_UAU.2
FIA_UID.2	None	N/A
FMT_MOF.1	FMT_SMR.1	FMT_SMR.1
	FMT_SMF.1	FMT_SMF.1
FMT_MSA.1	[FDP_ACC.1 or FDP_IFC.1]	FDP_ACC.1
	FMT_SMR.1	FMT_SMR.1
	FMT_SMF.1	FMT_SMF.1
FMT_MSA.3	FMT_MSA.1	FMT_MSA.1
	FMT_SMR.1	FMT_SMR.1
FMT_MTD.1	FMT_SMR.1	FMT_SMR.1

SFR	Dependencies	Fulfilled by SFRs in this ST
	FMT_SMF.1	FMT_SMF.1
FMT_SAE.1	FMT_SMR.1	FMT_SMR.1
	FPT_STM.1	This dependency is fulfilled by OE.TIME in the operational environment.
FMT_SMF.1	None	N/A
FMT_SMR.1	FIA_UID.1	FIA_UID.2
FPT_ITC.1	None	N/A
FPT_ITT.1	None	N/A
FTP_TRP.1	None	N/A
DDM_DIS.1	None	N/A
DDM_DEP.1	DDM_DIS.1	DDM_DIS.1

6.3.3 Security Assurance Requirements Rationale

EAL2+ was selected as the assurance level because the TOE is a commercial product whose users require a moderate level of independently assured security. The BMC Atrium Device and Dependency Mapping product is targeted at a relatively benign environment with good physical access security and competent administrators and users. Within such environments, it is assumed that attackers will have an attack potential that can be characterized as basic. BMC Atrium Device and Dependency Mapping provides a level of protection that is appropriate for operational environments that implement IT applications asset management. As such, it is believed that EAL2+ provides an appropriate level of assurance in the security functions offered by the TOE. ALC_FLR.2 was added to provide assurance that the vendor will respond to reports of new threats and reported defects in the TOE.

7 TOE SUMMARY SPECIFICATION

7.1 Mapping of the TSFs to SFRs

The specified TOE Security Functions (TSFs) work together to satisfy the TOE SFRs. Table 18 provides a mapping of SFRs to the TSFs to show that each SFR is captured within a security function.

Table 18. Mapping of TSFs to SFRs

TSF	SFR
Security Audit Data Generation	**FAU_GEN.1** Audit data generation
	FAU_GEN.2 User identity association
	FAU_SAR.1 Audit review
	FAU_SAR.2 Restricted audit review
	FAU_SAR.3 Selected audit review
	FAU_STG.2 Guarantees of audit data availability
	FAU_STG.3 Action in case of possible audit data loss
Cryptographic Support	**FCS_CKM.1(1)** Cryptographic key generation (RSA)
	FCS_CKM.1(2) Cryptographic key generation (AES)
	FCS_CKM.4 Cryptographic key destruction
	FCS_COP.1 (1) Cryptographic operation (AES)
	FCS_COP.1 (2) Cryptographic operation (RSA)
	FCS_COP.1 (3) Cryptographic operation (SHS)
	FCS_COP.1 (4) Cryptographic operation (HMAC)
	FPT_ITC.1 Inter-TSF confidentiality during transmission
	FPT_ITT.1 Basic internal TSF data transfer protection
	FTP_TRP.1 Trusted path
Security Management	**FDP_ACC.1** Subset access control
	FDP_ACF.1 Security attribute based access control
	FMT_MOF.1 Management of security functions behavior
	FMT_MSA.1 Management of security attributes
	FMT_MSA.3 Static attribute initialization
	FMT_MTD.1 Management of TSF data
	FMT_SAE.1 Time-limited authorisation
	FMT_SMF.1 Specification of management functions
	FMT_SMR.1 Security roles
Identification and Authentication	**FIA_AFL.1** Authentication failure handling
	FIA_ATD.1 User attribute definition
	FIA_SOS.1 Verification of secrets
	FIA_UAU.2 User authentication before any action
	FIA_UAU.7 Protected authentication feedback
	FIA_UID.2 User identification before any action
Discovery and Dependency Mapping	**DDM_DIS.1** Discovery
	DDM_DEP.1 Determine dependency relationships

7.2 Security Audit Data Generation

The TOE generates audit data that is recorded into an audit log. All log entries are marked with a reliable timestamp indicating the time and date of the entry and each entry indicates the type of event, the identity of the subject making the change, and the outcome of the event (success or failure). The timestamp is generated by the TOE using reliable time obtained from the environment. Each log file records the startup and end of logging activity. The composition of each audit record is described in Table 20 on page 47.

The TOE records all connection attempts and therefore can record unsuccessful login attempts, including the username provided.

To use the TOE's audit functionality, users must be logged in and a member of one of the security groups with read access listed in Table 21 on page 47.

To assure the availability of audit data, the TOE employs a disk space monitor. If there is less than an Administrator-prescribed amount of space on either of the disks, the monitor shuts the TOE down gracefully and prevents it from re-starting. If this occurs, a TOE Administrator must allocate more disk space to logging using the 'Manage ADDM Disks' functionality from the ADDM GUI before the TOE can be restarted. An administrator in the operational environment may have to make more disk space available to the TOE before this can occur.

To prevent the log files growing without bound, the TOE rolls the logs on a daily basis. Logs are automatically compressed seven days after they are created. Logs are compressed with the standard gzip tool. The compressed logs may be downloaded from the appliance and opened with any of a variety of tools that support gzip compression, such as Linux or MacOS gunzip, WinZip or 7-Zip on Windows. When a log is compressed it is no longer available in any of the log selector drop down lists. Compressed logs are automatically deleted 30 days after the initial log file was created. Old logs can be manually compressed or deleted in the log viewer, or directly using the command line. In the case of audit log exhaustion, it will be the most recent logs that are maintained, since older logs may have been automatically deleted after 30 days.

The TOE's OpenSSL cryptographic modules report the failure of encryption and decryption operations, including the operation type and the failure of key generation activity, to a log file. Additionally, these modules include a self-test functionality for the FIPS-certified AES algorithm in accordance with the FIPS requirement to periodically validate the encryption library. For the encryption algorithms AES and RSA, the self-test performs a known answer test (KAT). If the self test fails, the library goes into self-test failed mode and does not perform any other cryptographic function. In this case, attempts to connect to the UI and CLI will fail until the problem is corrected.

Audited events are grouped as shown in the following table:

Table 19. Audit Event Groups

Audit Event Group	Description	Audit Events
Appliance Config	Appliance configuration events	REBOOT_APPLIANCE_REQUEST
		RESTART_SERVICES_REQUEST
		SHUTDOWN_APPLIANCE_REQUEST
		ENABLE_MAINTENANCE_MODE
		DISABLE_MAINTENANCE_MODE
Audit Log	Audit log purge	AUDIT_LOG_PURGE_EVENT
Cluster	Cluster Management	CLUSTER_MGMT_CLUSTER_CHANGED
		CLUSTER_MGMT_COORDINATOR_CHANGED
		CLUSTER_MGMT_MEMBER_CHANGED
		CLUSTER_MGMT_MEMBER_JOINING
		CLUSTER_MGMT_MEMBER_LEAVING
		CLUSTER_MGMT_SERVICES_START
		CLUSTER_MGMT_SERVICES_STOP
		CLUSTER_MGMT_CHANGE_REJECTED
cmdb-export	Synchronization with the Configuration Management Database (if applicable)	CMDB_SYNC_BATCH
		CMDB_SYNC_BLACKOUT_ADD
		CMDB_SYNC_BLACKOUT_UPDATE

Audit Event Group	Description	Audit Events
		CMDB_SYNC_BLACKOUT_REMOVE
		CMDB_SYNC_START
		CMDB_SYNC_STOP
		CMDB_SYNC_DEVICE_FILTER_SAVE
		CMDB_SYNC_COMPONENT_FILTER_SAVE
		CMDB_SYNC_CI_FILTER_SAVE
Consolidation	Events related to the consolidation of scanned data from multiple devices	CONSOLIDATION_REGISTERED_TO_CA
		CONSOLIDATION_UNREGISTERED_FROM_CA
		CONSOLIDATION_APPROVED_DA
		CONSOLIDATION_RELEASED_DA
		CONSOLIDATION_SET_AS_CONSOLIDATION
		CONSOLIDATION_SET_AS_DISCOVERY
		CONSOLIDATION_REGISTERED_DA
		CONSOLIDATION_AUTOAPPROVED_DA
		CONSOLIDATION_REMOVED_DA
Datastore Edit	Changes to objects and the relationships between those objects that make up the data model	DATASTORE_ADD_NODE
		DATASTORE_MODIFY_NODE
		DATASTORE_REMOVE_NODE
		DATASTORE_ADD_RELATIONSHIP
		DATASTORE_REMOVE_RELATIONSHIP
		MASS_DELETE
DIP	Changes to Data Integration Point (DIP). DIPs handle communications with target systems.	DIP_CREATE_IP
		DIP_MODIFY_IP
		DIP_CREATE_CONNECTION
		DIP_MODIFY_CONNECTION
		DIP_CREATE_QUERY
		DIP_MODIFY_QUERY
		DIP_TEST_CONNECTION
		DIP_TEST_QUERY
		DIP_DELETE_IP
		DIP_DELETE_CONNECTION
		DIP_DELETE_QUERY
		DIP_JDBC_UPLOAD
		DIP_JDBC_FAILED_UPLOAD
Discovery Config	Changes to the configuration of the discovery function.	START_SCANNING_REQUEST
		STOP_SCANNING_REQUEST
		EMERGENCY_STOP_SCANNING_REQUEST
		ADD_SNAPSHOT_SCAN_EVENT
		REMOVE_SNAPSHOT_SCAN_EVENT
		ADD_SCHEDULED_SCAN_EVENT
		REMOVE_SCHEDULED_SCAN_EVENT
		UPDATE_SCHEDULED_SCAN_EVENT
		MODIFY_SCAN_LEVEL_EVENT
		MODIFY_EXCLUDE_RANGES_EVENT
		CANCEL_SCHEDULED_SCAN_EVENT
Discovery Ruleset	Change to the ruleset that defines the discovery configuration	MODIFY_DISCOVERY_CONFIG_OPTION
Security	Changes to users and groups	SECURITY_ADD_USER

Audit Event Group	Description	Audit Events
		SECURITY_MODIFY_USER
		SECURITY_DELETE_USER
		SECURITY_USER_CHANGE_PASSWORD
		SECURITY_ADD_GROUP
		SECURITY_DELETE_GROUP
		SECURITY_GROUP_PERMISSIONS_MODIFY
		SECURITY_OPTIONS_MODIFY
		SECURITY_USER_STATE_MODIFY
		SECURITY_ADD_LDAP_GROUP
		SECURITY_EDIT_LDAP_GROUP
		SECURITY_DELETE_LDAP_GROUP
Windows proxy	Changes to the Windows Proxy configuration	WINDOWS_PROXY_LOGLEVEL_CHANGE
		WINDOWS_PROXY_SIGN_CONF_FILE
		WINDOWS_PROXY_CONF_FILE_UPLOAD
		WINDOWS_PROXY_CONF_FILE_UPLOAD_FAILURE
		WINDOWS_PROXY_READ_NO_PERMISSION
		WINDOWS_PROXY_MODIFY_NO_PERMISSION
		WINDOWS_PROXY_PROCESS_TERMINATE
		WINDOWS_PROXY_SETTINGS_UPDATE
		WINDOWS_PROXY_CONF_FILE_RELOAD
		WINDOWS_PROXY_CONF_FILE_RELOAD_FAILURE
		WINDOWS_PROXY_THROW_OVERLOADED
Search	Searches of data using the UI	SEARCH_QUERY
		SEARCH_CANCELLED
ECA Reasoning	Changes associated with the Event Condition Action (ECA) engine, which processes the rules that are generated from patterns, used to perform the discovery actions.	ECA_START_ENGINE
		ECA_STOP_ENGINE
		ECA_ADD_EVENT
		ECA_REMOVE_EVENT
		ECA_CHANGE_SCANLEVELS
		ECA_COMMIT_PATTERN_STATE
		ECA_EXECUTE_PATTERN
		ECA_MODIFY_PATTERN_CONFIG
		ECA_ADD_PATTERN_PACKAGE
		ECA_ACTIVATE_PATTERN_PACKAGE
		ECA_DEACTIVATE_PATTERN_PACKAGE
		ECA_DELETE_PATTERN_PACKAGE
		ECA_ENABLED_PATTERN_MODULE
		ECA_DISABLED_PATTERN_MODULE
		ECA_ACTIVATE_PATTERN_MODULE
		ECA_DEACTIVATE_PATTERN_MODULE
		ECA_DELETE_PATTERN_MODULE
		ECA_REPLACE_PATTERN_MODULE
		ECA_ADD_PATTERN_MODULES
UI Access	Access to the UI	LOGON_EVENT
		LOGON_FAILURE_EVENT
		LOGOFF_EVENT

The individual events that belong to these groups are available to authorized users via the Audit page in the UI.

Table 20. Audit Record Composition

Name	Description
Event	The type of event
Event Group	The event group to which the event belongs. The purpose of the event group is to provide a filter for viewing related event types.
User	The user ID who initiated the event
Full Name	The full name of the user who initiated the event
User Groups	The name of the group(s) the user who initiated the event belongs
When	When the event was logged. The TOE records the following date and time information for each audit record using the host virtual machine's time of day clock: ■ Day ■ Month ■ Year ■ Hours ■ Minutes
Summary	Summary description of the event

Table 21. Audit Record Review Access

Security Group	Audit Review
admin	Full access to audit records (admin, read, write, purge)
system	Full access to audit records (admin, read, write, purge)
appmodel	Purge
cmdb-export-administrator	No access
discovery	Purge
public	No access
readonly	No access
unlocker	No access

Access to the audit log is controlled by providing users (via their membership(s) in security groups) with the read, write, purge, and admin audit permissions detailed in Table 22.

Table 22. Audit Permissions Definitions

Permission	Definition
Read	The user role is provided read access to the audit log. The audit log is viewable directly from the CLI or via the log viewer option in the UI. When using the UI to view the audit log, the authorized user can search using the following criteria: ■ From: the start date and time of the search. The default for the "from" field is 24 hours prior to when the UI page is loaded. ■ To: the end time and date of the search. The default for searching the "to" field is to display Day Month Year Hours Minutes meaning that the audit log can be search up to the current time.

Permission	Definition
	■ User ID: a filter can be set to search only for events logged to a specific user ID. ■ Event Group: a drop-down filter is available to search only for events belonging to a particular event group or category. The event group provides a means for viewing related event types. ■ Events: a drop-down filter is available to search only for events of a specific type. Audit records can also be exported in comma separated variable (CSV) format so that a spreadsheet or text editor can be used for more detailed searching and sorting of the audit log.
Write	Audit records are written to the audit log for actions taken by user roles having this permission.
Purge	The user role is able to purge (delete) the audit log of all events older than one month. Notes: 1. The TOE will not permit audit events less than one month old to be deleted. 2. Purging is an auditable event. Following a purge operation, the newest audit event in the audit log will be a record of the purge.
Admin	The user role can configure and administer the TOE's audit service.

The functionality described above satisfies the FAU_GEN.1, FAU_GEN.2, FAU_SAR.1, FAU_SAR.2, FAU_SAR.3, FAU_STG.2, and FAU_STG.3 functional requirements by recording the occurrence of security relevant modifications to the TOE and protecting this information from access or modification by unauthorized users.

7.3 Cryptographic Support

The TOE includes two FIPS 140-2 validated cryptographic modules (Table 23) which are used by the TSF to provide cryptographic services to the TOE.

- The first (Red Hat Enterprise Linux 6.6 OpenSSL Cryptographic Module) is a user space cryptographic library that provides the following FIPS 140-2 validated crypto services to applications on the ADDM Appliance TOE component:
  - cryptographic key generation and destruction;
  - AES encryption/decryption [FIPS 197];
  - RSA digital signature standard [FIPS 186-2] signing and verification (RSA),
  - Secure Hash Standard (SHS – [FIPS 180-3]) hashing, and
  - Keyed Hash Message Authentication Code (HMAC – [FIPS 198-1]) for message integrity.

  In support of the above, the TSF generates 2048-bit asymmetric keys in accordance with [ANSI X9.31]. It also uses a FIPS approved RNG in accordance with [ANSI X9.31] to generate 256-bit AES keys.

- The second validated module (OpenSSL FIPS Object Module) is also a user-space cryptographic library that provides the same FIPS 140-2 validated cryptographic services described above to Windows Proxy TOE components.

Table 23. FIPS 140-2 Validated Cryptographic Modules

TOE Component	Cryptographic Module Name	Overall Security Level	Certificate Number
ADDM Appliance	Red Hat Enterprise Linux 6.6 OpenSSL Cryptographic Module	1	2441
Windows Proxy	OpenSSL FIPS Object Module (Software Version: 2.0, 2.0.1, 2.0.2, 2.0.3, 2.0.4 or 2.0.5) Note: The version used by the TOE is 2.0.	1	1747

Table 24 identifies the algorithms implemented by the TOE which have been awarded algorithm certificate numbers by the CAVP:

Table 24. TOE Algorithm Implementations

Module	Algorithm	Certificate	Usage	Keys / CSPs	Modes Used by TSF	Comment(s)
Red Hat Enterprise Linux 6.6 OpenSSL Cryptographic Module	AES	3104, 3105, 3106,3107, 3108, 3109, 3110, 3111, 3112, 3113, 3114, 3119	encryption, decryption	256 bits	CBC	Meets FIPS 197. Used for TLS connections between TOE components (ADDM Appliance and Windows Proxy) as well as between the ADDM Appliance and any Scanning BMC Atrium Discovery Appliances located on the network. Used for HTTPS (i.e., HTTP over TLS) connections between the ADDM Appliance TOE component and the management console. Used for SSH connections between the ADDM Appliance TOE component and the management console.
OpenSSL FIPS Object Module		1884				
Red Hat Enterprise Linux 6.6 OpenSSL Cryptographic Module	RSA	1583, 1584, 1586, 1590	sign, verify, key generation	2048 bits	2048-bit key generation 2048-bit signature generation and verification	Meets: ■ FIPS 186-2 ■ ANSI X9.31 ■ PKCS#1 v1.5 Used for set-up of TLS connections between TOE components (ADDM Appliance and Windows Proxy) as well as between the ADDM Appliance and any Scanning BMC Atrium Discovery Appliances on the network. Used to set-up HTTPS (i.e., HTTP over TLS) connections between the ADDM Appliance TOE component and the management console. Used for the set-up of SSH
OpenSSL FIPS Object Module		960				

Module	Algorithm	Certificate	Usage	Keys / CSPs	Modes Used by TSF	Comment(s)
						connections between the ADDM Appliance TOE component and the management console.
Red Hat Enterprise Linux 6.6 OpenSSL Cryptographic Module	SHS	2547,2563, 2564,2565, 2566,2567, 2568,2569, 2570,2574, 2575,2577	Message digest (hash)	160 bits	Message digest created on Byte-only data	Meets FIPS 180-3 Used within the TLS, including HTTPS, and SSH protocols.
OpenSSL FIPS Object Module		1655				
Red Hat Enterprise Linux 6.6 OpenSSL Cryptographic Module	HMAC	1944, 1945, 1946, 1947, 1948, 1949, 1950, 1951, 1955, 1956, 1958	Keyed-hash message authentication code (message integrity)	Key size (KS): 160 bits Byte size (BS): 20	Message integrity during TLS handshake	Meets FIPS 198-1 and FIPS 180-3 Used within the TLS, including HTTPS, and SSH protocols.
OpenSSL FIPS Object Module		1126				
Red Hat Enterprise Linux 6.6 OpenSSL Cryptographic Module	RNG	3116, 3117, 3118, 3119, 3120, 3121, 3122, 3123, 3124, 3125, 3126	AES keys	128, 192, 256 bits	N/A	Meets ANSI X9.31 Used to generate the symmetric AES keys used within the TLS, HTTPS, and SSH protocols.
OpenSSL FIPS Object Module		985				

The TSF uses TLSv1.0 to secure the communications channels between:

- any Scanning BMC Atrium Discovery Appliances (trusted IT products) on the network and the TOE's Appliance component. The appliances are identified through configuration of the connection; and
- the TOE's Appliance and Windows Proxy components.

The TLS v1.0 protocol includes functionality for data encryption, server authentication, message integrity, and client authentication. In the evaluated configuration, the TOE employs the TLS_DHE_RSA_WITH_AES_256_CBC_SHA cipher suite for TLS-secured channels:

- Ephemeral Diffie-Hellman (DH) key exchange with DH parameters signed by an RSA certificate (DHE_RSA) to provide the following benefits:
  - the TOE component can be authenticated by the IT entity it is exchanging keys with, and
  - this mechanism provides perfect forward secrecy for the TLS session key
- 256-bit AES block encryption algorithm
- CBC (Cipher Block Chaining) mode
- SHA1 HMAC construction for integrity protection

The TSF secures the browser-based UI communications for remote users in the IT environment and the ADDM Appliance TOE component using HTTP over TLSv1.0 (represented as HTTPS in this Security Target). In the evaluated configuration, this trusted path is secured using the TLS_DHE_RSA_WITH_AES_256_CBC_SHA cipher suite described above. The client is identified via username and password.

The TSF also implements the Secure Shell transport layer protocol (SSH) as specified in RFC 4253. SSH is used to establish a secure channel for CLI access to the ADDM Appliance TOE component for remote TOE Administrators on the management console. The TOE includes the OpenSSH

package, configured to use the FIPS 140-2 Red Hat Enterprise Linux 6.6 OpenSSL Cryptographic Module to the extent summarized in Table 24. The client is identified via username and password.

The Cryptographic Support function enforces the FCS_CKM.1(1), FCS_CKM.1(2), FCS_CKM.4, FCS_COP.1(1), FCS_COP.1(2), FCS_COP.1(3), FCS_COP.1(4), FPT_ITC.1, FPT_ITT.1 and FTP_TRP.1 requirements as summarized below:

- FCS_CKM.1(1), FCS_CKM.1(2), and FCS_CKM.4 are implemented by FIPS 140-2 validated cryptographic modules. Key generation and destruction is carried out by the modules to support the set-up and teardown of TLS and SSH-based secured communication channels between either TOE components or the TOE and external IT entities.
- Cryptographic operations (FCS_COP.1(1), FCS_COP.1(2), FCS_COP.1(3), and FCS_COP.1(4)) required by the TLS (including HTTP over TLS) and SSH protocols are carried out by the validated cryptographic modules.
- FPT_ITC.1 is enforced by the TSF through the use of TLS to secure the communications channel between the ADDM Appliance TOE component and any Scanning BMC Atrium Discovery Appliances located on the network.
- FPT_ITT.1 is enforced by the TSF through the use of TLS to secure all communications between the ADDM Appliance TOE component and each Windows Proxy TOE component located on the network.
- FTP_TRP.1 is enforced by the TSF by ensuring that:
  - remote administrators and users can only access the TOE's browser-based UI over a TLS-secured HTTP connection; and
  - remote administrators can only access the TOE's CLI via an SSH-secured channel.

7.4 User Data Protection

The TOE has two broad classes of user: user and administrator. User permissions are determined by each user's membership in Security Groups as described at Table 25. Note that only four default Security Groups (*readonly*, *public*, *system*, and *admin*) can login to the TOE and access the GUI.

The TSF makes use of the Security Groups to enforce access to TOE objects by TOE users (i.e., enforcement of the BMC Atrium Discovery Access Control SFP). The BMC Atrium Discovery Access Control Security Function Policy (SFP) is implemented in the ADDM Appliance Subsystem, and controls access to the GUI, the CLI and datastore and the datastore partitions, Discovery credentials, audit log, CMDB export data and TOE configuration settings. Non-administrative users in either the *public* or *readonly* Security Group have restricted access to TOE functionality. This satisfies the requirements of FDP_ACC.1 and FDP_ACF.1.

Table 25 – TOE User Roles

Role	Security Group	Description
Administrator	admin and system	Users in these two security groups are TOE Administrators who have full access to the TOE (via both the GUI and the CLI).
	cmdb-export-administrator	Users in this security group are TOE users with limited TOE administrative rights. They have access to all TSF export-related data. This security group can build, modify, delete and run Exporters.
	discovery	Users in this security group are TOE users with limited TOE administrative rights. They have access to all of the TOE's discovery-related data. This user can start and stop discovery, add and remove credentials, and enable or disable audit logging.
	unlocker	Users in this security group are TOE users with limited TOE administrative rights. They are able to unlock and unblock user accounts that have been locked or blocked after exceeding the number of permitted authentication failures
	appmodel	Users in this security group are limited TOE administrators. They can write and edit (discovery) patterns, and create nodes to model business applications. They cannot view credentials but can run discovery in order to test patterns. This user can only access the TOE via the browser-based GUI.
User	public	Users in this security group are TOE users. They have read/write access to TOE general functionality. They do not have access to discovery credentials.
	readonly	Users in this security group are limited TOE users. They have read access to TOE general functionality. They cannot view the credentials for logging into target hosts (discovery credentials).

7.5 Identification and Authentication

The Identification and Authentication function enforces the FIA_AFL.1, FIA_ATD.1, FIA_SOS.1, FIA_UAU.2, FIA_UAU.7, FIA_UID.2 requirements as described below.

The TOE Administrator is responsible for initializing users on the system, including defining their initial access credentials and their security settings. During this initial set-up, users are assigned:

- a user name
- a password, and
- membership in at least one Security Group. All users of the TOE must be a member of one or more Security Groups. Membership in Security Groups (refer to Section 7.6 - Security Management for additional information re: Security Groups) defines the functionality that a TOE user is entitled to access.

Non administrative TOE users are restricted to logging into the TOE via its browser-based UI as described below:

- The user browses to the secure (i.e., https) URL provided by the TOE Administrator and enters the provided user name and password. When users authenticate, the TOE provides only limited feedback in the form of the number of characters typed appearing as asterisks during authentication.
- If the password is entered incorrectly, the user is not provided access to any TOE functionality and is re-prompted up to the maximum number of tries defined by the Account Blocking setting defined for that user (Table 27). When this limit has been reached, the user account will be blocked from accessing the TOE's login page until at least after a period of time has elapsed that corresponds with the Automatically Unblock setting applied to that user account (Table 27).
- Upon first successful logon instance, the user may be required to change his/her password consistent with the password complexity rules described at Table 26.
- After confirming the new password by entering it a second time, the TOE user is directed to a home page that displays only the TOE functionality available to the user based on the security groups to which his/her user name is associated with.

TOE login functionality via the browser-based UI is the same for TOE Administrators as described above. TOE Administrators also have CLI access to a specific subset of TOE functionality over an SSH-secured channel from the management console. When a TOE Administrator logs into the CLI, no TSF functionality is provided until he/she is successfully identified by user name and authenticated through the entry of the correct password. When entering the password at the CLI, the TOE provides no feedback to the user regarding the number of characters being entered; all keystroke feedback information is suppressed. The TOE Administrator can configure password quality metrics for CLI authentication per Table 26

Table 26. Administrator-specified Password Quality Metrics

Password Quality Metric	Description	TSF Default
Minimum password length	Administrators can select a minimum length (1 to 32 characters) or selecting "None" to enforce no minimum length.	The default setting is 6 characters.
Minimum password re-use history	Administrators can specify a password history length (3, 5, 10, or 20) to prevent users from recycling passwords too quickly, or select "None" to enforce no restrictions on password reuse.	The default setting is 10.
Password complexity	Administrators can select the following constraints (complexity rules) for the passwords: ■ Must contain uppercase characters [setting: true or false] ■ Must contain lowercase characters [setting: true or false] ■ Must contain numeric characters [setting: true or false]	All constraints enabled (i.e., set to true).

Password Quality Metric	Description	TSF Default
	■ Must contain special characters [setting: true or false] ■ Must not contain sequences (e.g., AAA, ppp, or 222) [setting: true or false]	

The password complexity rules described in Table 26 result in a password that requires a minimum of six (6) characters consisting of at least one of each of the following:

- one lower case character,
- one upper case character,
- one numeric character, and
- one special character.

Table 27. User Authentication Failure Handling

Failure Mechanism	Description	TSF Default
Account Blocking	Administrators can specify the number of unsuccessful login attempts (1, 2, 3, 4, or 5) before an account is blocked, or select "Never" to not block accounts.	The default is 3 tries prior to blocking.
Automatically Unblock	Administrators can specify either: ■ the period (4, 5, 10, 15, 20, 30 or 60 minutes) after which the a blocked account will be automatically unblocked, or ■ "Never" to prevent automatically unblocking. Note: The guidance documentation cautions the TOE Administrator that if "…*you select Never, there is a chance that you lock out the system account*".	The default setting is 10 minutes

7.6 Security Management

The Security Management function enforces the FDP_ACC.1, FDP_ACF.1, FMT_MOF.1, FMT_MSA.1, FMT_MSA.3, FMT_MTD.1, FMT_SAE.1, FMT_SMF.1, FMT_SMR.1 requirements as described below.

The TOE has two broad classes of user (user and administrator) that is controlled by each user's membership in Security Groups as described at Table 28. TOE Administrators who are members of the *admin* or *system* Security Group are able to configure users to use specific parts of the TOE's functionality. Users are allocated a user name and a password, which they must enter in order to log in to the system. Each user is a member of one or more Security Groups, which defines the parts of the system that user is permitted to access. By default all new users are created as members of the *public* Security Group.

The TSF makes use of the above-noted Security Groups and a series of specific Security Permissions to enforce access to TOE objects on TOE users (i.e., enforcement of the BMC Atrium Discovery Access Control SFP). Default users in either the *public* or *readonly* Security Group have restricted access to TOE functionality as described below in Table 28.

Table 28. TOE User Roles

Role	Security Group	Description
Administrator	admin & system	Users in these two security groups are TOE Administrators who have full access to the TOE (via both the UI and the CLI).

Role	Security Group	Description
	cmdb-export-administrator	Users in this security group are TOE users with limited TOE administrative rights. They have access to all TSF export-related data. This security group can build, modify, delete and run Exporters.
	discovery	Users in this security group are TOE users with limited TOE administrative rights. They have access to all of the TOE's discovery-related data. This user can start and stop discovery, add and remove credentials, and enable or disable audit logging.
	unlocker	Users in this security group are TOE users with limited TOE administrative rights. They are able to unlock and unblock user accounts that have been locked or blocked after exceeding the number of permitted authentication failures (See Table 27 in Section 7.4 at page 53 for additional information).
	appmodel	Users in this security group are limited TOE administrators. They can write and edit (discovery) patterns, and create nodes to model business applications. They cannot view credentials but can run discovery in order to test patterns. This user can only access the TOE via the browser-based UI.
User	public	Users in this security group are TOE users. They have read/write access to the TOE discovery and dependency mapping functionality, but do not have access to discovery credentials.
	readonly	Users in this security group are limited TOE users. They have read access to TOE general functionality. They cannot view the credentials for logging into target hosts (discovery credentials).

Table 29 summarizes the administration operations provided by the TSF. These operations are enabled by membership in the *admin* or *system* Security Group, which in term is defined by 86 specific Security Permissions in 11 broad categories (administration, appliance, application server, baseline, consolidation, discovery, model, reasoning, security, user interface and vault) applied to the user account.

Table 29. Summary of TSF Administration Operations

Category	Operations
Appliance	Configuration. Control: restart services, reboot or shutdown the Appliance, and put the Appliance into maintenance mode. Logs: manage logs and log levels. Backup & Restore: operate the Appliance backup feature that enables the administrator to back up the Appliance. Baseline status: check and configure the Appliance baseline. Performance: view charts showing the Appliance performance over the last 30 days. Support Services: create an archive of diagnostic information. Miscellaneous: configure miscellaneous settings. JDBC Drivers: configure JDBC drivers that are available for the TOE.
Discovery	Platforms: configure the commands used for each target operating system. Sensitive Data Filters: mask any sensitive data which may otherwise be seen in the command output. Vault Management: manage the Credential Vault Device Capture: capture an SNMP device and dump the MIB of an SNMP agent, which is then used to request that support be included in the TOE for that SNMP device.
Model	View Taxonomy: view the system taxonomy. Model Maintenance: configure model maintenance settings. Custom Categories: set up data categories. Search Management: view any search in progress and cancel searches.
Security	Users: add, remove and modify users. Groups: add, remove and modify groups. Security Policy: configure the Appliance security options. HTTPS: configure the HTTPS settings for the Appliance. Active Sessions: view all users who are currently logged in. Audit: configure the Appliance's audit feature.

TOE Administrators in the listed Security Groups can manage TSF data as described in Table 30.

Table 30. Management of TSF Data

TSF Data	Security Group	Operation	Default
Audit logs	*admin* *system*	TOE Administrators who are members of the identified Security Groups can configure the TOE to permit purging (deleting) of the audit log up to the current date. These TOE Administrators can purge the audit log via the Purge item in the Administration tab of the UI by selecting a date range	Purging is enabled for audit log records older than one month prior to the current date Default selectable age of logged events to be purged are: ■ 1 month ago, ■ 3 months ago, ■ 6 months ago, ■ 12 months ago, and ■ 24 months ago.
	admin *system*	TOE Administrators who are members of the identified Security Groups can query (search) the audit log using any of the following fields: ■ From (start date and time of search) ■ To (end time and date of search) ■ User Name (filter to search for events logged to a specific user) ■ Event Group (filter to search for events belonging to specific Event Group ■ Events (filter to search for events of a specific type)	■ From: 24 hours prior to current TOE time ■ To: current TOE time
	admin *system*	TOE Administrators who are members of the identified Security Groups can export audit log events by selecting the “Export as CSV” item in the Administration tab of the UI.	N/A
Security Groups	*admin* *system*	TOE Administrators who are members of the identified Security Groups can: ■ list (i.e., query) the details of all current Security Groups in the TOE configuration ■ modify default Security Group names and modules that users can access ■ create new Security Groups ■ delete any Administrator-created Security Groups ■ add users to and remove users from Security Groups. Changes made above do not take effect on users until the next time users in the affected Security Group(s) log in.	N/A
Security Permissions	*admin* *system*	TOE Administrators who are members of the identified Security Groups can view (i.e., query) Security Permissions by Security Group or by User. These same users can add or remove Security Permissions to Security Groups or users as needed.	The default configuration provides admin and system users with full access to all facets of the TOE’s functionality.

TSF Data	Security Group	Operation	Default
Discovery Patterns	*admin* *appmodel* *discovery* *system*	TOE users who are members of the identified Security Groups can view (i.e., query) and modify either default or custom-created Discovery Patterns.	N/A
	admin *appmodel* *system*	In addition to the above, TOE users who are members of the identified Security Groups can also upload new patterns, modify the entire pattern source, and download (i.e., export) patterns for offline storage or review via the Pattern Management UI. These users can run the `tw_pattern_management` utility via the CLI to: ■ upload patterns to the Appliance, ■ activate or deactivate patterns on the Appliance, and ■ delete Patterns from the Appliance that are no longer required.	N/A
TOE Configuration Settings	*admin* *system*	TOE Administrators in the identified Security Groups can ■ view (query) and edit (modify) the identification of the Appliance (name, description, and email address of the person or group responsible for administering the TOE) ■ view (query) and edit (modify) the TOE"s network interface and routing settings, including name resolution settings ■ view (query) and edit (modify) the TOE's email settings ■ view (query) and edit (modify) the amount of memory allocated to individual JVMs up to a limit of 1024 MB ■ turn-on (change default) and configure (modify) usage data collection (i.e., control how much information, if any, is sent to BMC about the TOE's usage. ■ localize the appliance by setting (modify) keyboard layout, timezone, and time – including synchronization with an NTP service.	The following are default TOE configuration settings: ■ Network Interface Card (NIC) is configured to use DHCP to obtain an IP address ■ 512MB memory allocation per JVM ■ data usage submission is disabled by default.
TOE security policy settings	*admin* *system*	TOE users in the identified Security Groups can configure (change default and modify) the following security options for users: ■ password quality metrics (see Table 26 on page 52) ■ force password change to new users upon initial logon	The referenced tables to the left identify default security policy settings where applicable. ■ Force password change for new users upon initial logon is not enabled. ■ login page has no text in the legal notice banner ■ login page autocomplete set to UI cannot be

TSF Data	Security Group	Operation	Default
		■ account blocking after a specified number of authentication failures (see Table 27 on page 53) ■ define if, and when unused accounts are deactivated (see Table 31 on page 57) ■ appearance of the login page (define legal notice banner message, allow browser autocomplete) ■ UI defined to be incorporated as part of an umbrella UI (to prevent cross site framing or "clickjacking" attacks) ■ UI configured to only use HTTPS	used as part of an umbrella page (i.e., set to "yes") ■ By default HTTP access is enabled and HTTPS access is disabled. HTTPS must be enabled in the evaluated configuration.

Authorized TOE Administrators determine which TOE users can read the Audit log by either:

- including the user in a Security Group that includes the permission */model/audit/read*, or
- specifically adding this Security Permission to the user's profile.

The BMC Atrium Discovery Access Control SFP provides granular access to the TSF by users through the application of Security Permissions. For example, to log in to the TOE, not only must a user have been created in the TSF, but the user must be in a Security Group that has at least the following three Security Permissions:

- *security/user/passwd*,
- *appserver/login*, and
- *appserver/module/home*.

By default, only the four following Security Groups have this permission:

- *readonly*,
- *public*,
- *system*, and
- *admin*

Security Permissions enforced by the TSF are additive. When a user is granted Security Permissions by a TOE Administrator (by adding the user to a Security Group), the permissions associated with that group are added to the user's existing permissions. Each user has a token which is assigned by the security subsystem in the TOE. Whenever a privilege is requested by a user, the security service checks the database to see if that particular user has permission to carry out that particular task.

TOE Administrators who are members of either the *admin* or *system* Security Group can specify the authentication failure thresholds and resultant TSF actions listed at Table 27. These TOE Administrators can also specify the time-limited authorizations listed in Table 31.

Table 31. Specification of Time-limited Authorizations

	Description	TSF Default
Password Expiry Period	Administrators can specify the life of passwords (30, 45, 75, 90 105, and 120 days) before they are automatically expired or they can select "None" to enforce no expiry period.	The default setting is 90 days
Password Expiry Warning	Administrators can configure the TOE to warn users that their password will soon expire when they login to the user interface	The default setting is 10 days.

	Description	TSF Default
	from a list of: never, 5, 10, and 15 days. If "never" is selected, users will receive no warning of impending password expiration. Note that the expiry warning period cannot be set to more than the Password Expiry Period.	
Account Deactivation	Administrators can specify either: ■ the period (15, 30, 45, 60, 75, 90, 105, and 120 days) after which unused user accounts are deactivated, or ■ "Never" to not deactivate accounts.	not applicable
Disabled Accounts can be Reactivated	Administrators can specify (Yes or No) if accounts can be reactivated. Accounts can only be reactivated by a TOE Administrator.	No. This must be changed to 'Yes' for the evaluated configuration.

7.7 Protection of the TSF

The ADDM Appliance Subsystem uses the cryptographic functionality described in Section 7.3 to satisfy the requirements for Protection of the TSF.

7.7.1.1.1 Inter-TSF Confidentiality During Transmission

The TSF uses TLS v1.0 to secure the communications channels between the ADDM Appliance Subsystem and any other ADDM Appliance (trusted IT product) on the network. The TLS v1.0 protocol includes functionality for data encryption, server authentication, message integrity, and client authentication. In the evaluated configuration, the TOE employs the TLS_DHE_RSA_WITH_AES_256_CBC_SHA cipher suite. This protects data from unauthorized disclosure during transmission, and satisfies the requirements for FPT_ITC.1.

7.7.1.1.2 Basic Internal TSF Data Transfer Protection

The TSF uses TLS v1.0 to secure the communications channels between the ADDM Appliance Subsystem and the Active Directory Windows Proxy Subsystem, and between the ADDM Appliance Subsystem and the Credential Windows Proxy Subsystem. The TLS v1.0 protocol includes functionality for data encryption, server authentication, message integrity, and client authentication. In the evaluated configuration, the TOE employs the TLS_DHE_RSA_WITH_AES_256_CBC_SHA cipher suite. This protects data from disclosure and modification as it is transmitted between parts of the TOE and satisfies the requirements for FPT_ITT.1.

7.8 Trusted Path/Channel

The ADDM Appliance Subsystem secures the browser-based GUI communications between remote users in the IT environment and the ADDM Appliance TOE component using HTTP over TLSv1.0. In the evaluated configuration, this trusted path is secured using the TLS_DHE_RSA_WITH_AES_256_CBC_SHA cipher suite.

The TSF also implements the Secure Shell transport layer protocol (SSH) as specified in RFC 4253. SSH is used to establish a secure channel for CLI access to the ADDM Appliance TOE component for remote TOE Administrators on the management console. The TOE includes the OpenSSH package, configured to use the FIPS 140-2 Red Hat Enterprise Linux 6.6 OpenSSL Cryptographic Module as described in Section 7.3.

Only remote users are able to initiate calls to the ADDM Appliance Subsystem, and are assured of the correct end point by entering the address of the ADDM Appliance. The ADDM Appliance Subsystem only allows users to access via a TLS v1.0 secured HTTP connection or an SSH-secured channel. Users must be identified and authenticated before being allowed access to TOE functions. This satisfies the requirements of FTP_TRP.1.

7.9 Discovery and Dependency Mapping

The Discovery and Dependency Mapping function enforces the DDM_DIS.1 and DDM_DEP.1 requirements as described below.

The TOE provides authorized users with the means to automatically discover IT components consisting of servers (physical and virtual), network devices, and applications running on components on the end-user organization's network. Authorized users are presented with a graphical user

interface (Discovery UI) that provides each user with the means to simply manage the discovery process. Authorized users can define scanning levels for individual discovery runs. As part of this set-up, the user can define discovery scans over a range of IP addresses as well as define IP ranges to exclude from a discovery run. Discovery runs are easily started and stopped, and when stopped, the runs are paused so that their state is saved thereby enabling runs to continue when they are later restarted.

Through the discovery process, the TOE builds a topology of the organization's applications and infrastructure, including servers, operating systems, software, network devices, business applications, and identifies their dependencies. After discovering the components and relationships of business applications, the TOE uses patterns to automatically build an application dependency map based on that information. It also continuously scans for changes in the components and relationships and updates the dependency map accordingly.

The TOE's event-based reasoning engine populates the data model that describes the applications and IT components' dependency relationships. The TOE makes use of patterns and a pattern language to create, add, and delete items in the data model. A pattern is a sequence of commands written in the pattern language, which contain instructions that identify scanned entities which are then used to create the data model. The pattern language describes how software instances and business application instances are determined by grouping nodes and sequencing commands. The purpose of the pattern language is to abstract the complexity of the Event/Condition/Action (ECA) rule engine and present a simple interface for non-programmers. These patterns drive the instruction of the reasoning engine.

Prior to initiating a discovery and dependency mapping scan, the authorized user can configure the following settings:

- Port settings: The port settings are used to identify TCP/UDP ports on target hosts that will be scanned.
- Device identification settings: These settings determine the mechanisms that the TOE will use identify devices on the network.
- Session settings: During a discovery scan, the TOE attempts to access host systems to obtain details of processes running. Credentials including IDs and passwords, and credential-like entities (Windows proxies and SNMP credentials) for different access methods, can be stored on the TOE to allow the required level of access. The session settings define how the discovery scan will employ sessions to login and run commands on target hosts.
- Scanning settings: These settings define how the scan will be carried out by the TOE. In general, there are two broad types of scans:
  - Sweep scans: The TOE will try to determine what is at each endpoint in a scan range. It attempts to log in to each found device to determine the device type.
  - Full discovery: The TOE retrieves all the default information for found hosts.
- SQL integration settings: The TOE will use these settings to connect to databases.
- Other discovery settings: The TOE will exercise additional discovery options, such as including desktops in the discovery process (or not) subject to the authorized user's requirements.


bmcsoftware
BUSINESS RUNS ON I.T.™