SECURITY TARGET

CITADEL HERCULES® AUTOMATED VULNERABILITY REMEDIATION

VERSION 2.2.0

Document No. 1451-011-D001

Version 1.13, 27 February 2004

Prepared for:

Citadel Security Software Inc.
8750 N. Central Expressway
Suite 100
Dallas Texas 75231

Prepared by:

Electronic Warfare Associates-Canada, Ltd.
55 Metcalfe St., Suite 1600
Ottawa, Ontario
K1P 6L5

Security Target

Citadel Hercules® Automated Vulnerability Remediation

Version 2.2.0

Document No. 1451-011-D001

Version 1.13, 27 February 2004

<Original> Approved by:

Project Engineer:	G. Gibbs	
Project Manager:	E. Connor	
Program Director:	P. Zatychec	
	(Signature)	(Date)

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

1.1 GENERAL

This introductory section presents security target (ST) identification information, an overview of the product and an overview of the ST structure. A brief discussion of the ST development methodology is also provided.

An ST document provides the basis for the evaluation of an information technology (IT) product or system under the Common Criteria for Information Technology Security Evaluation (CC). Within the ST the product or system which is being evaluated is referred to as the Target of Evaluation (TOE). An ST principally defines:

- A set of assumptions about the security aspects of the environment, a list of threats which the product is intended to counter, and any known rules with which the product must comply (see Section 3, Security Environment).
- A set of security objectives and a set of security requirements are presented in Sections 4 and 5, Security Objectives and IT Security Requirements, respectively.
- The IT security functions provided by the TOE which meet that set of requirements (see Section 6, TOE Summary Specification).

The structure and contents of this ST comply with the requirements specified in the CC, Part 1, Annex C and Part 3, Chapter 5.

1.2 IDENTIFICATION

Title:	Citadel Hercules Automated Vulnerability Remediation Version 2.2.0 Security Target
Registration:	383-4-18
Common Criteria Conformance Claim	The TOE is CC Part 2 conformant and CC Part 3 conformant.
Evaluation Assurance Level (EAL):	The TOE is EAL 3 conformant.
Protection Profile Conformance:	The TOE does not claim conformance with any Protection Profile (PP).
Common Criteria Identification:	Common Criteria for Information Technology Security Evaluation, Version 2.1, August 1999, with all current approved interpretations.
International Standard:	ISO/IEC 15408:1999

Authors: This document has been written by EWA-Canada on behalf of Citadel Security Software Inc.

1.3 PRODUCT OVERVIEW

The Hercules® AVR is a network security administration tool that is intended to be used in conjunction with advanced network vulnerability assessments.

The purpose of the product is to enable the deliberate and controlled remote, automated vulnerability remediation (AVR) of all classes of identified network vulnerabilities on large-scale enterprise level Windows® and Unix (Solaris/Linux) based networks.

Hercules® AVR provides network security administrators with the ability to prioritize and remediate vulnerabilities using automated fixes that have been developed, tested, verified as being correct and validated as being appropriate, by trusted and dedicated IT security professionals.

New vulnerabilities are being discovered on a daily basis. It has been estimated that it takes approximately one manhour of labour to manually correct one vulnerability on one client machine. For all but the smallest networks, manually correcting vulnerabilities imposes an unacceptable workload and cost for valuable and often scarce network and security administration resources. The Hercules® AVR product overcomes this problem with Automated Vulnerability Remediation (AVR). Hercules® AVR offers the following significant features:

- Interoperability – Hercules® AVR supports many industry leading vulnerability assessment scanners.
- Multi-tiered Architecture – The Hercules® AVR Administrator Console can be configured to manage multiple Hercules® AVR Servers.
- Administrator Control – Administrators maintain complete control over the selection of which vulnerabilities are to be remediated.
- Multiple O/S Support – In addition to Windows® platforms, Hercules® AVR supports UNIX (Sun Solaris) and Linux (Red Hat).
- Reporting – Detailed reports organize the vulnerability remediation data and can be used to measure the ongoing success of frequent vulnerability remediation cycles.
- Consistent Remediation – Hercules® AVR provides a consistent method of remediation across an entire network, it does not depend on the skill level of individual technicians when resolving vulnerabilities.
- Device Grouping – Administrators can place devices into logical groups and schedule remediation by groups.

- Roll-back Capabilities – Administrators have the ability to roll-back system changes and patch installations when necessary.

- V-Flash – Administrators can stay current on the latest vulnerability remediation signatures through the Hercules® AVR V-Flash update service.

- Remediation Policies – Users can define remediation policies for a single device or group of devices.

- Best Practices – Hercules® AVR offers complete support for the 'best practices' of vulnerability remediation.

At a high level, Hercules® AVR is designed to:

- Aggregate vulnerability and remediation information from leading sources including SecurityFocus, BugTraq, CERTs and other internet sources.

- Import scan information from vulnerability scanners and combine this information to perform remediation from a single source.

- Create profiles and remediation signatures that match scanner-independent vulnerability information and client machines with their corresponding remediations.

- Allow an administrator to target network machines for automated remediation.

- Support CVE compliance by displaying CVE identifiers and supporting searching using these identifiers.

Fundamentally, the Hercules® AVR product provides enterprise administrators with the ability to manage a large-scale vulnerability remediation process in a manner that is both systematic and comprehensive. Today many organizations employ an incomplete hybrid of manual and partially automated techniques that are often implemented in an ad-hoc manner. Hercules® AVR is a tool that is intended to bring a defined and systematic maturity into these security-critical processes.

In a Windows® environment, Hercules® AVR is a product that provides and includes all of the functionality typically associated with the vulnerability remediation capabilities of commercial and open source vulnerability scanners. These typically provide registry fixes for Windows® machines. However, this type of vulnerability only represents a small sub-set of the vulnerabilities that require remediation. The Hercules® AVR product expands this set to include the automated remediation of vulnerabilities associated with the following five classes of vulnerabilities:

- **Software Defects** – Hot fixes, patches, registry settings, etc.

- **Unnecessary/Insecure Services** – Telnet, Remote Access, FTP etc.

- **Insecure Accounts** – Null Passwords, Admin No Password, etc.

- **Back Doors** – NetBus, BackOrifice, SubSeven etc.

- **Mis-Configurations** – NetBIOS, file system privileges, Null Sessions etc.

The Hercules® AVR product is designed to operate on standard TCP/IP networks and can remediate vulnerabilities on both Windows® and UNIX (Solaris/Linux) based clients.

The Hercules® AVR human machine interface (HMI) provides the user with complete control over the functionality of the product. The HMI allows the user to specify:

- An automated frequency with which client systems will request updated vulnerability remediations.

- Manual remediations for selected client machines.

- Specific vulnerabilities which will not be remediated.

1.4 CONVENTIONS, TERMINOLOGY AND ACRONYMS

This section identifies the formatting conventions used to convey additional information and terminology having specific meaning. It also defines the meanings of abbreviations and acronyms used throughout the remainder of this document

1.4.1 Conventions

This section describes the conventions used to denote CC operations on security requirements and to distinguish text with special meaning. The notation, formatting and conventions used in this ST are largely consistent with those used in the CC. Selection presentation choices are discussed here to aid the ST reader.

The CC allows several operations to be performed on functional requirements; *assignment*, *iteration*, *refinement* and *selection* are defined in paragraph 148 of Part 1 of the CC.

- The assignment operation is used to assign a specific value to an unspecified parameter, such as the length of a password. An assignment is indicated by showing the value in italicised text within square brackets [assignment: *values*].

- The refinement operation is used to add detail to a requirement, and thus further restricts a requirement. Refinement of security requirements is denoted by **bold** text. There are no refinements within this ST.

- The selection operation is used to select one or more options provided by the CC in stating a requirement. Selections are denoted by italicised text within square brackets [selection: *value(s)*].

- The iteration operation is used to apply a security functional requirement to more than one aspect of the TOE. Iterations are denoted by repeating the text of the security functional requirement for each of the applicable aspects of the TOE.

1.4.2 Terms

This section describes the terms that are used throughout this ST. When possible, terms are defined as the exist in the CC.

Assets	Information or resources to be protected by the countermeasures of a TOE.
Attack	An attempt to bypass security controls on an IT System. The attack may alter, release or deny data. Whether an attack will succeed depends on the vulnerability of the IT System and the effectiveness of existing countermeasures.
Audit	The independent examination of records and activities to ensure compliance with established controls, policy and operational procedures and to recommend indicated changes in controls, policy or procedures.
Audit Trail	In an IT System, a chronological record of system resource usage, this includes user login, file access or other activities and whether any actual or attempted security violations occurred, legitimate and unauthorised.
Authentication	To establish the validity of a claimed user or object.
Availability	Assuring information and communications services will be ready for use when expected.
Compromise	An intrusion into an IT System where unauthorised disclosure, modification or destruction of sensitive information may have occurred.
Confidentiality	Assuring information will be kept secret, with access limited to appropriate persons.
Evaluation	Assessment of a PP, a ST or a TOE, against defined criteria.
Information Technology (IT) System	May range from a computer system to a computer network.
Integrity	Assuring information will not be accidentally or maliciously altered or destroyed.
IT Product	A package of IT software, firmware and/or hardware providing functionality designed for use or incorporation within a multiplicity of systems.
Network	Two or more machines interconnected for communications.

Protection Profile (PP)	An implementation independent set of security requirements for a category of TOE that meet specific consumer needs.
Security	A condition that results from the establishment and maintenance of protective measures that ensure a state of inviolability from hostile acts or influences.
Security Policy	The set of laws, rules and practices that regulate how an organisation manages, protects and distributes sensitive information.
Security Target (ST)	A set of security requirements and specification to be used as the basis for evaluation of an identified TOE.
Target of Evaluation (TOE)	An IT product or system and its associated administrator and user guidance documentation that is the subject of an evaluation.
Threat	The means through which the ability or intent of a threat agent to adversely affect an automated system, facility or operation can be manifest. A potential violation of security.
TOE Security Functions (TSF)	A set of all hardware, software and firmware of the TOE that must be relied upon for the correct enforcement of the TSP.
TOE Security Policy (TSP)	A set of rules that regulate how assets are managed, protected and distributed within a TOE.
TSF Data	Data created by and for the TOE that might affect the operation of the TOE.
TSF Scope of Control	The set of interactions that can occur with or within a TOE and are subject to the rules of the TSP.
User	An entity (human user or external IT entity) outside of the TOE that interacts with the TOE.
Vulnerability	Hardware, firmware or software flaw that leaves an IT System open for potential exploitation. A weakness in automated system security procedures, administrative controls, physical layout, internal controls and so forth that could be exploited by a threat to gain unauthorised access to information, unauthorised privileges or disrupt critical processing.

1.4.3 Acronyms

AVR	Automatic Vulnerability Remediation
CC	Common Criteria for Information Technology Security Evaluation
CERT	Computer Emergency Response Team
CM	Configuration Management
CVE	Common Vulnerabilities and Exposures
EAL	Evaluation Assurance Level
HMI	Human Machine Interface
IT	Information Technology
O/S	Operating System
SSH	Secure Shell
SSL	Secure Sockets Layer
ST	Security Target
TCP/IP	Transmission Control Protocol / Internet Protocol
TOE	Target of Evaluation
TSC	TOE Scope of Control
TSF	TOE Security Functions
TSP	TOE Security Policy

2 TARGET OF EVALUATION DESCRIPTION

2.1 EVALUATED CONFIGURATION

The Hercules® AVR product is designed to facilitate the automatic vulnerability remediation of devices on a network. The product imports vulnerability information from a number of third party, commercial vulnerability scanner products and consolidates this information into a single view of the vulnerabilities of each device in the network. The product provides a sequence of automatically executable remediation steps known as a 'remediation signature' which will correct each recognized vulnerability. Users of the product may download new signatures from the 'V-flash' server operated by Citadel Security Software. The Hercules® AVR product provides an interface which allows users to view the listed vulnerabilities of devices on the network. Logical groupings of devices may be defined. An automatic remediation schedule may be defined for a group. In addition, a specific list of vulnerabilities to be remediated, known as a 'remediation profile' may be defined for the group. The evaluated configuration of the Hercules® AVR Version 2.2.0 product (build 1792 with software update V-flash Version 202000501, dated 2 Feb 2004) consists of:

a. The Hercules® AVR Administrator Console executing on an Intel® Pentium based PC running Windows® 2000 Server with all service packs, Windows® 2000 Advanced Server with all service packs, Windows® XP Professional with all service packs, Windows® 2003 Standard Edition or Windows® 2003 Enterprise Edition as the operating system. Internet Explorer 5.5 or above is also required. The minimum hardware requirements for the Hercules® AVR Administrator Console are specified in the Citadel Hercules® AVR Automated Vulnerability Remediation Installation Guide. The required setup of the Hercules® AVR Administrator Console is described in the Hercules® AVR Security Configuration Guide.

b. One or more Hercules® AVR Server(s) executing on an Intel® Pentium based PC running Windows® 2000 Server with Service Pack 4, Windows® 2000 Advanced Server with Service Pack 4, Windows® 2003 Standard Edition or Windows® 2003 Enterprise Edition as the operating system. For the Windows® 2000 server family IIS 5.0 is also required. For the Windows® Server 2003 family IIS 6.0 is also required. Internet Explorer 6.0 with service pack 1 is required for all installations. The minimum hardware requirements for a Hercules® AVR Server are specified in the Citadel Hercules® AVR Automated Vulnerability Remediation Installation Guide. The required setup of a Hercules® AVR Server is described in the Hercules® AVR Security Configuration Guide.

c. One or more network devices with Hercules® AVR Client Version 2.2.0 installed on a supported Windows® operating system. The supported versions of the Windows® operating system are Windows® NT 4.0 Workstation with service pack 6, Windows® NT 4.0 Standard Server with service pack 6, Windows® NT 4.0 Terminal Server with service pack 6, Windows® 2000 Professional with any service pack, Windows® 2000 Server with any service pack, Windows® 2000 Advanced Server with any service pack, Windows® XP Professional with any SP, Windows® Server 2003 Standard Edition and Windows® Server 2003 Enterprise Edition. For Windows® NT 4.0

platforms, Internet Explorer 5.5 with service pack 2 or above is also required. The minimum system requirements for Windows® Clients are specified in the Citadel Hercules® AVR Automated Vulnerability Remediation Installation Guide.

d. One or more network devices with Hercules® AVR Client Version 2.2.0 installed on a supported version of the UNIX operating system. The supported versions of the UNIX operating system are Solaris (SPARC) 2.6, 7, 8, 9 and Red Hat (Intel) 6.0, 6.1, 6.2, 7.0, 7.1, 7.2, 7.3, 8, 9. The minimum system requirements for UNIX Clients are specified in the Citadel Hercules® AVR Automated Vulnerability Remediation Installation Guide.

2.2 TOE BOUNDARY

The Hercules® AVR product consists of the following major components:

- The ***Hercules® AVR Administrator Console*** provides the HMI for the product. It uses SSL-based communications with the Hercules® AVR Server(s), and has the ability to interact with Windows® user accounts, domain privileges and NTFS privileges. It authenticates (using Windows® integrated authentication) to Internet Information Server on the Hercules® AVR server. The Hercules® AVR Administrator Console is designed to be installed and used on a trusted and appropriately configured and controlled Windows® machine that is used for network administration. Users of the Hercules® AVR Administrator Console require full administrative privileges on the machine running the console as well as the Hercules® AVR Server and all client machines. The Hercules® AVR Administrator Console provides the HMI for the product and includes the display and input devices through which the user interacts with the Hercules® AVR application.

- The ***Hercules® AVR Server*** is a Windows® service that communicates with the Hercules® AVR Client to distribute remediation profiles and gather remediation progress data. Multiple Hercules® AVR Servers may be deployed within a network and administered from a single Hercules® AVR Administrator Console. The Hercules® AVR Server is designed to be installed and used on a trusted and appropriately configured and controlled Windows® server.

- The ***Hercules® AVR Windows® Clients*** are services that perform remediation activities on client machines. The clients establish HTTPS/SSL-based communication to the Hercules® AVR Server.

- The ***Hercules® AVR Unix Clients,*** provide functionality which is equivalent to Windows® client capabilities. Unix clients require a root account to install, configure, and execute Unix daemons, use of Unix file system access control and the use of ssh for installation.

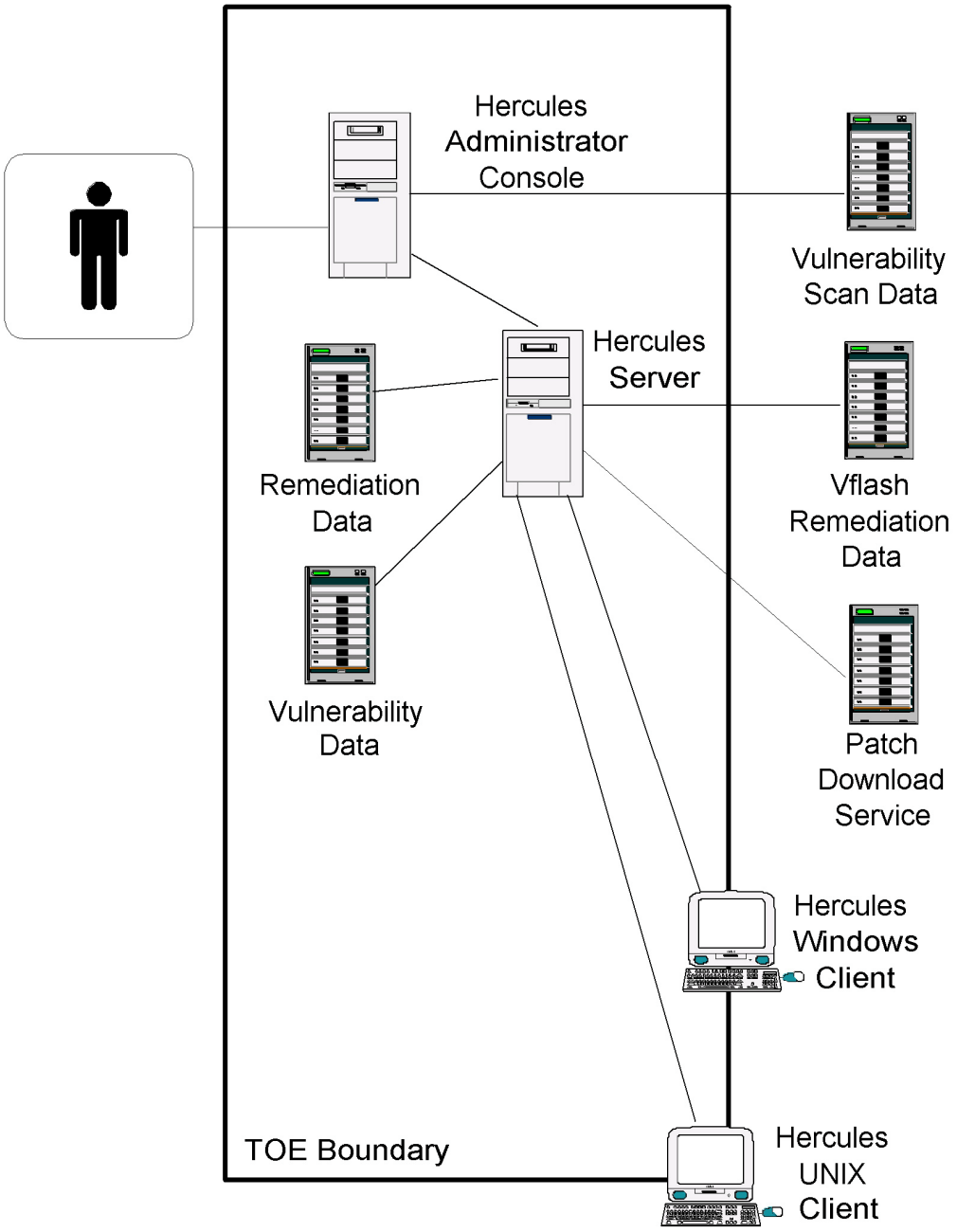

Figure 1 TOE Boundary Diagram

The Hercules® AVR Administrator Console and all of the Hercules® AVR Servers fall within the TOE Boundary as do the data stores associated with the Remediation and Vulnerability data used by the Hercules® AVR Server. Figure 1 shows that client machines are partially inside the TOE Boundary and partially outside the boundary. The parts of the client machines which fall within the TOE Boundary consist of the Hercules® AVR Client software and those

portions of the operating system necessary to provide authentication and secure communications with the Hercules® AVR Server.

The Hercules® AVR product is designed for the use of network administrators and it is assumed that these users are appropriately trained and experienced. Further, it is assumed that the user does not have malicious intent and configures the product and its host platforms in accordance with the guidance documentation. The product will not prevent a user from carelessly configuring or using the Hercules® AVR such that network protection is compromised.

3 TOE SECURITY ENVIRONMENT

3.1 ASSUMPTIONS

The following conditions are assumed to exist in the operational environment:

A.BACKUP	The organization operating the TOE has good backup and recovery procedures which are followed; allowing the TOE to be recovered to a secure configuration after a hardware failure.
A.CMS	In an environment where the Hercules® AVR client software is installed by remote means on Windows client using the Hercules® AVR Client Management Services (CMS), the server and clients are assumed to reside on a protected network.
A.CONFIG	The servers running the Remediation Server and the Administrator Console have been configured securely as described in the Guidance documents and are maintained in that secure configuration. In particular: a. They are configured with the minimal operating system features installed and / or enabled to permit operation of the TOE. b. They are configured with minimal system privileges. c. They are configured with user accounts for authorized system administrators only and do not provide any end user accounts.
A.GOODOS	The Operating System of the client machines has been configured in accordance with the Hercules® AVR Security Configuration Guide and therefore may be trusted to function correctly for those OS functions required by the TOE component that is installed on the client machine.
A.KNOWLEDGE	TOE Users have knowledge of the Windows® 2000/XP/2003 operating systems, networking technology and general IT security practices.
A.NOEVIL	TOE Users are non hostile and follow all guidance documents.
A.PHYSICAL	The Server and Administrator elements of the TOE are physically secure and only authorized personnel have physical access to these elements of the TOE.
A.TOEUSER	There is only one category of TOE user. All authorized TOE users have full access to all of the TOE's functions and for this reason there is no distinction between TOE users and TOE administrators. For the remainder of this document the phrase 'TOE User' shall be employed.

3.2 THREATS

The threats discussed below are addressed by a compliant TOE. The threat agents are either human users or external IT entities not authorized to use the TOE. Additionally, threat agents may be users with administrative privileges that introduce vulnerabilities (either deliberately or inadvertently) by misconfiguring network systems from a security perspective. Threat agents are assumed to have a low level of sophistication, but may have knowledge of vulnerabilities and access to attack methods which are in the public domain. The TOE is not designed to withstand attack by sophisticated, highly motivated or well funded threat agents. The assets that are subject to attack are the components of the TOE itself and / or the resources of the client systems protected by the TOE.

T.BADDATA	A network attacker may attempt to provide the Remediation Server with erroneous remediation information in an attempt to compromise the Client systems.
T.CLIENT	An unauthorized person may have administrator / root control of one of the client systems and may use that control to attempt to compromise the Remediation Server.
T.CONSOLE	A network attacker may attempt to gain control of the TOE through the Hercules® AVR Administration Console.
T.EXPLOIT	A network attacker may attempt to exploit vulnerabilities on a client system protected by the TOE in order to gain unauthorized access to the resources of the client system.
T.NETEXPLOIT	A network attacker may attempt to exploit vulnerabilities on a client system protected by the TOE in an attempt to compromise other network resources.
T.REMSERVER	A network attacker may attempt to gain control of the Hercules® AVR Remediation Server
T.SNIFF	A network attacker may intercept and monitor communications between the Remediation Server and the Client systems and use the information gained to compromise the Remediation Server and / or a Client system.
T.SNIFFSCAN	A network attacker may monitor communications between the Remediation Server and a vulnerability scanner to learn vulnerabilities of client systems.
T.SPOOF	A network attacker may attempt to imitate the Remediation Server and provide erroneous remediation information to a client system in order to compromise the client.
T.SPOOFCLIENT	A network attacker may attempt to imitate a client system in order to gain information about the vulnerabilities of the client system.
T.SPOOFSCAN	A network attacker may attempt to provide the Remediation Server with erroneous vulnerability assessment information in an attempt to

prevent the remediation of vulnerable network systems.

3.3 ORGANIZATIONAL SECURITY POLICIES

There is no requirement for the TOE to comply with any organizational security policy statements or rules.

4 SECURITY OBJECTIVES

4.1 SECURITY OBJECTIVES FOR THE TOE

O.CLIENTAUTH	The TOE must provide a mechanism for a two way authentication between client systems and the Remediation Server.
O.CLIENTPROT	The TOE must protect itself against attacks initiated by client systems.
O.CLIENTREM	The TOE must provide effective remediation of known and reported vulnerabilities for client systems.
O.HMI	The TOE must provide a controlled interface to its functionality such that only authorized TOE users are able to access the interface.
O.NETATK	The TOE must protect itself against network attackers.
O.REMDATA	The TOE must ensure that its remediation data is obtained from trusted sources and must provide a mechanism to ensure the integrity of this data.
O.SCANDATA	The TOE must ensure that its scanner data is obtained from trusted sources and must provide a mechanism to ensure the confidentiality and integrity of this data.

4.2 ENVIRONMENT SECURITY OBJECTIVES

The list below details the security objectives for the environment in which the TOE resides. These objectives are to be met through the application of procedural and / or administrative measures. They do not impose any additional security requirements upon the TOE.

OE.AUTHUSER	Only authorized personnel are permitted physical access to the TOE.
OE.BACKUP	Good backup and recovery procedures for the TOE must be in place.
OE.GOODOS	Those portions of the client operating system required for the correct operation of the TOE must function correctly.
OE.GOODUSER	Knowledgeable, non malicious users with system administrator privileges must be assigned to install, configure, administer, operate and maintain the TOE.
OE.GUIDANCE	The administrator(s) responsible for the TOE must ensure that the TOE is installed, configured, administered and operated in accordance with the guidance documents.
OE.SECURECOM	The network on which the TOE resides must protect the confidentiality and integrity of information exchanged between the distributed elements of the TOE when client machines are initially installed remotely using the Hercules® AVR Client Management Service (CMS).

5 IT SECURITY REQUIREMENTS

5.1 TOE SECURITY REQUIREMENTS

Section 5 provides security functional and assurance requirements that must be satisfied by a compliant TOE. These requirements consist of functional components from Part 2 of the CC and an Evaluation Assurance Level (EAL) containing assurance components from Part 3 of the CC.

5.2 TOE SECURITY FUNCTIONAL REQUIREMENTS

The security functional requirements for this ST consist of the following components from Part 2 of the CC, summarized in Table 1.

CC Part 2 Security Functional Components	
Identifier	**Name**
FAU_GEN.1	Audit Data Generation
FAU_SAR.1	Audit Review
FAU_SEL.1	Selective Audit
FDP_IFC.1	Subset Information Flow Control
FDP_IFF.1	Simple Security Attributes
FDP_ITC.1	Import of User Data without Security Attributes
FDP_ITT.1	Basic Internal Transfer Protection
FDP_ROL.1	Basic Rollback
FIA_AFL.1	Authentication Failure Handling
FIA_SOS.1	Verification of Secrets
FIA_UAU.2	User Authentication Before Any Action
FIA_UID.2	User Identification Before Any Action
FMT_MSA.1	Management of Security Attributes
FMT_MSA.3	Static Attribute Initialization
FMT_MTD.1	Management of TSF Data
FMT_SMF.1	Specification of Management Functions
FMT_SMR.1	Security Roles
FPT_ITT.1	Basic Internal TSF Data Transfer Protection
FPT_RVM.1	Non-bypassability of the TSP
FPT_SEP.1	TSF Domain Separation

CC Part 2 Security Functional Components	
Identifier	**Name**
FPT_STM.1	Reliable Time Stamps

Table 1 Summary of CC Part 2 Security Functional Requirements

FAU_GEN.1 Audit data generation

Hierarchical to: No other components.

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 [selection: *not specified*] level of audit; and
c) [assignment: *use of the Hercules® AVR Client Management Service, Patch Download Service or Vflash Service events in addition to the audit capabilities of the underlying operating system*].

FAU_GEN.1.2 The TSF shall record within each audit record at least the following information:
a) Date and time of the event, type of event, subject identity, and the outcome (success or failure) of the event; and
b) For each audit event type, based on the auditable event definitions of the functional components included in the ST, [assignment: *no other audit relevant information*]

Dependencies: FPT_STM.1 Reliable time stamps

FAU_SAR.1 Audit review

This component will provide authorised users the capability to obtain and interpret the information. In case of human users this information needs to be in a human understandable presentation. In case of external IT entities the information needs to be unambiguously represented in an electronic fashion.

Hierarchical to: No other components.

FAU_SAR.1.1 The TSF shall provide [assignment: *all TOE users*] with the capability to read [assignment: *all audit 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.

Dependencies:	FAU_GEN.1 Audit data generation
FAU_SEL.1	Selective audit
Hierarchical to:	No other components.
FAU_SEL.1.1	The TSF shall be able to include or exclude auditable events from the set of audited events based on the following attributes: a) [selection: *event type*] b) [assignment: *client machine identification*].
Dependencies:	FAU_GEN.1 Audit data generation FMT_MTD.1 Management of TSF data
FDP_IFC.1	Subset information flow control
Hierarchical to:	No other components.
FDP_IFC.1.1	a. The TSF shall enforce the [assignment: *SERVER_SFP*] on [assignment: *Hercules® AVR Servers and client machines when the client machine requests a remediation profile from a Hercules® AVR Server*]. b. The TSF shall enforce the [assignment: *IMPORT_SFP*] on [assignment: *Hercules® AVR Servers when importing vulnerability scan data and vulnerability remediation data from outside the TOE boundary*].
Dependencies:	FDP_IFF.1 Simple security attributes
FDP_IFF.1	Simple security attributes
Hierarchical to:	No other components.
FDP_IFF.1.1	a. The TSF shall enforce the [assignment: *SERVER_SFP*] based on the following types of subject and information security attributes: [assignment: *(1) Identification and authentication of the client machine; and (2) format of client machine remediation status information*]. b. The TSF shall enforce the [assignment: *IMPORT_SFP*] based on the following types of subject and information security attributes:

[assignment: *(1) The identification and authentication of the TOE user and Vflash server; and (2) the format of the source data*].

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

(1) For the transfer of a remediation signature from the Hercules® AVR Server to a client machine; (a) the requesting client machine has been identified as authorised by the server using either certificates or in the absence of certificates the IP Address, Domain Name or NETBIOS name; and (b) the format of the client machine remediation status information is recognized.

(2) For the import of Vulnerability Scan data to the server; (a) the file to be imported has been specified by the authorized TOE User; and (b) the file meets the format expected by the TOE for the file purpose.

(3) For the import of remediation data the Hercules® Vflash server is successfully authenticated by the Hercules® AVR Server using either certificates or in the absence of certificates the IP Address, Domain Name or NETBIOS name.].

FDP_IFF.1.3 The TSF shall enforce the [assignment: *no additional information flow control SFP rule*s].

FDP_IFF.1.4 The TSF shall provide the following [assignment: *no additional SFP capabilitie*s].

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

FDP_IFF.1.6 The TSF shall explicitly deny an information flow based on the following rules: [assignment: *none*].

Dependencies: FDP_IFC.1 Subset information flow control

FMT_MSA.3 Static attribute initialisation

FDP_ITC.1 Import of user data without security attributes

Hierarchical to: No other components.

FDP_ITC.1.1	The TSF shall enforce the [assignment: *IMPORT_SFP*] when importing user data, controlled under the SFP, from outside of the TSC.
FDP_ITC.1.2	The TSF shall ignore any security attributes associated with the user data when imported from outside the TSC.
FDP_ITC.1.3	The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TSC: [assignment: *no additional importation control rule*s].
Dependencies:	FDP_IFC.1 Subset information flow control
	FMT_MSA.3 Static attribute initialisation

FDP_ITT.1	Basic internal transfer protection
Hierarchical to:	No other components.
FDP_ITT.1.1	The TSF shall enforce the [assignment: *SERVER_SFP*] to prevent the [selection: *disclosure, modification*] of user data when it is transmitted between physically-separated parts of the TOE.
Dependencies:	FDP_IFC.1 Subset information flow control

FDP_ROL.1	Basic rollback
Hierarchical to:	No other components.
FDP_ROL.1.1	The TSF shall enforce [assignment: *SERVER_SFP*] to permit the rollback of the [assignment: *automatic vulnerability remediations*] on the [assignment: *client machines*].
FDP_ROL.1.2	The TSF shall permit operations to be rolled back within the [assignment: *time period between the completion of the remediation that is to be rolled back and the start of the next remediation*].
Dependencies:	FDP_IFC.1 Subset information flow control

FIA_AFL.1	Authentication failure handling
Hierarchical to:	No other components.

FIA_AFL.1.1	The TSF shall detect when [assignment: *a user configurable number, with an unlimited default value, of*] unsuccessful authentication attempts occur related to [assignment: *consecutive unsuccessful authentication attempts since the last successful authentication to the Hercules® AVR Administrator Console or Hercules® AVR Server*].
FIA_AFL.1.2	When the defined number of unsuccessful authentication attempts has been met or surpassed, the TSF shall [assignment: *lock the account attempting to log in and generate an audit record. The account shall remain locked until unlocked by an authorised Hercules® AVR User.*]
Dependencies:	FIA_UAU.1 Timing of authentication
FIA_SOS.1	Verification of Secrets
Hierarchical to:	No other components
FIA_SOS.1.1	(1) The TSF shall provide a mechanism to verify that secrets meet [assignment: *the requirements that user passwords are a minimum of 8 characters in length, include a combination of alphanumeric, special, upper and lower case character, and are changed at least once every 42 days*]. (2) The TSF shall provide a mechanism to verify that secrets meet [assignment: *the requirements that the Internal Hercules CMS Domain Administrator Password is a minimum of 8 characters in length, a maximum of 15 characters in length with each character generated randomly*].
Dependencies:	No dependencies
FIA_UAU.2	User authentication before any action
Hierarchical to:	FIA_UAU.1
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.
Dependencies:	FIA_UID.1 Timing of identification
FIA_UID.2	User identification before any action
Hierarchical to:	FIA_UID.1

FIA_UID.2.1	The TSF shall require each user to identify itself before allowing any other TSF-mediated actions on behalf of that user.
Dependencies:	No dependencies.

FMT_MSA.1	Management of security attributes
Hierarchical to:	No other components.
FMT_MSA.1.1	a. The TSF shall enforce the [assignment: *SERVER_SFP*] to restrict the ability to [selection: *create, modify, delete,* [assignment: *none*]] the security attributes [assignment: *identification and authentication of client machine*] to [assignment: *authorised Hercules® AVR Users*]. b. The TSF shall enforce the [assignment: *IMPORT_SFP*] to restrict the ability to [selection: *create, modify, delete,* [assignment: *none*]] the security attributes [assignment: *identification and authentication of client machine and Vflash server*] to [assignment: *authorised Hercules® AVR Users*].
Dependencies:	FDP_IFC.1 Subset information flow control FMT_SMF.1 Specification of management functions FMT_SMR.1 Security roles

FMT_MSA.3	Static attribute initialisation
Hierarchical to:	No other components.
FMT_MSA.3.1	The TSF shall enforce the [assignment: *SERVER_SFP and IMPORT_SFP*] to provide [selection: *permissive*] default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2	The TSF shall allow the [assignment: *authorised TOE users*] to specify alternative initial values to override the default values when an object or information is created.
Dependencies:	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles.

FMT_MTD.1	Management of TSF data
Hierarchical to:	No other components.
FMT_MTD.1.1	The TSF shall restrict the ability to [selection: *modify, delete,* [assignment: *aggregate, display*]] the [assignment: *vulnerability data, remediation data and client system vulnerability and remediation status*] to [assignment: *authorised Hercules® AVR users*].
Dependencies:	FMT_SMF.1 Specification of management functions FMT_SMR.1 Security roles

FMT_SMF.1	Specification of Management Functions
Hierarchical to:	No other components.
FMT_SMF.1.1	The TSF shall be capable of performing the following security management functions: [assignment: *a. specifying a list of client systems which are to be subject to automatic vulnerability remediation;* *b. specifying which vulnerabilities are to be remediated;* *c. scheduling automatic vulnerability remediations; and* *d. rolling back previously completed remediations*].
Dependencies:	No dependencies

FMT_SMR.1	Security roles
Hierarchical to:	No other components.
FMT_SMR.1.1	The TSF shall maintain the roles [assignment: *Hercules® AVR User and any other installation specific roles created by authorised Hercules® AVR Users*].
FMT_SMR.1.2	The TSF shall be able to associate users with roles.
Dependencies:	FIA_UID.1 Timing of identification

FPT_ITT.1	Basic internal TSF data transfer protection
Hierarchical to:	No other components.

FPT_ITT.1.1	The TSF shall protect TSF data from [selection: *disclosure, modification*] when it is transmitted between separate parts of the TOE.
Dependencies:	No dependencies

FPT_RVM.1	Non-bypassability of the TSP
Hierarchical to:	No other components.
FPT_RVM.1.1	The TSF shall ensure that TSP enforcement functions are invoked and succeed before each function within the TSC is allowed to proceed.
Dependencies:	No dependencies.

FPT_SEP.1	TSF domain separation
Hierarchical to:	No other components.
FPT_SEP.1.1	The TSF shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects.
FPT_SEP.1.2	The TSF shall enforce separation between the security domains of subjects in the TSC.
Dependencies:	No dependencies.

FPT_STM.1	Reliable time stamps
Hierarchical to:	No other components.
FPT_STM.1.1	The TSF shall be able to provide reliable time stamps for its own use.
Dependencies:	No dependencies.

5.3 SECURITY FUNCTIONAL REQUIREMENTS PROVIDED BY THE OPERATING SYSTEM

The Hercules® AVR product relies upon the underlying operating system to provide some of the security features of the product. Of the security functional requirements listed in the previous paragraph, the operating systems provides all or part of the functionality for these

functional requirements; FAU_GEN.1, FAU_SAR.1, FAU_SEL.1, FDP_ITT.1, FIA_AFL.1, FIA_SOS.1, FIA_UAU.2, FIA_UID.2, FMT_MSA.1, FMT_MSA.3, FMT_SMR.1, FPT_ITT.1, FPT_RVM.1, FPT_SEP.1 and FPT_STM.1.

5.4 INFORMATION FLOW CONTROL SECURITY FUNCTIONAL POLICIES

5.4.1 Hercules® AVR Server to Client Information Flow Control Security Functional Policy (SERVER_SFP)

The operating environment for the TOE consists of a Hercules® AVR Administrator Console and one or more Hercules® AVR Servers connected in a network with a number of client machines. It is expected that the client machines will contain vulnerabilities which will be automatically remediated by the Hercules® AVR Server on a scheduled basis. In an environment where the client machines are assumed to contain vulnerabilities the possibility always exists that one or more of the client machines have been compromised and may act maliciously towards the TOE. For this reason the only information that a Hercules® AVR Server will accept from any client machine is: (a) the identification of the client machine for authentication purposes when requesting a scheduled remediation, and (b) remediation status information during the course of a remediation session. All other information flow between the Hercules® AVR Server and a Hercules® AVR Client will consist of remediation profiles or rollback instructions sent from the Server to the client.

5.4.2 Vulnerability Scanner Import Information Flow Control Security Functional Policy (IMPORT_SFP)

The TOE relies upon data generated by one or more third party vulnerability scanner products in order to identify the vulnerabilities which exist on client machines. These scanner products fall outside the boundary of the TOE. The data generated by the scanners is also initially outside the TOE boundary. However, authorised TOE users, may import data from one of the recognised scanner products across the TOE boundary. If the vulnerability data is selected by an authorised TOE user and conforms to the expected format of data from one of the supported third party scanner products, then the TOE accepts that data as valid vulnerability information.

During the operation of the TOE the update of vulnerability remediation data must be performed on a regular basis. These updates are obtained from the trusted Hercules® AVR V-Flash server which falls outside the TOE boundary. The TOE uses SSL to ensure the fidelity of the data downloaded from the V-Flash server.

5.5 TOE SECURITY ASSURANCE REQUIREMENTS

The security assurance requirements for the TOE, comprise the requirements corresponding to the EAL 3 level of assurance as defined in the CC Part 3. The assurance components are summarized in the following table:

Assurance Class	Assurance Components	
	Identifier	**Name**
Configuration Management	ACM_CAP.3	Authorization controls
	ACM_SCP.1	TOE CM coverage
Delivery and Operation	ADO_DEL.1	Delivery procedures
	ADO_IGS.1	Installation, generation, and start-up procedures
Development	ADV_FSP.1	Informal functional specification
	ADV_HLD.2	Security enforcing high-level design
	ADV_RCR.1	Informal correspondence demonstration
Guidance Documents	AGD_ADM.1	Administrator guidance
	AGD_USR.1	User guidance
Life Cycle Support	ALC_DVS.1	Identification of security measures
Tests	ATE_COV.2	Analysis of coverage
	ATE_DPT.1	Testing: high-level design
	ATE_FUN.1	Functional testing
	ATE_IND.2	Independent testing – sample
Vulnerability Assessment	AVA_MSU.1	Examination of guidance
	AVA_SOF.1	Strength of TOE security function evaluation
	AVA_VLA.1	Developer vulnerability analysis

Table 2 EAL 3 Assurance Requirements

ACM_CAP.3 Authorization controls

Objectives

A unique reference is required to ensure that there is no ambiguity in terms of which instance of the TOE is being evaluated. Labeling the TOE with its reference ensures that users of the TOE can be aware of which instance of the TOE they are using.

Unique identification of the configuration items leads to a clearer understanding of the composition of the TOE, which in turn helps to determine those items which are subject to the evaluation requirements for the TOE.

Providing controls to ensure that unauthorized modifications are not made to the TOE, and ensuring proper functionality and use of the CM system, helps to maintain the integrity of the TOE.

Dependencies:

ALC_DVS.1	Identification of security measures

Developer action elements:

ACM_CAP.3.1D	The developer shall provide a reference for the TOE.
ACM_CAP.3.2D	The developer shall use a CM system.
ACM_CAP.3.3D	The developer shall provide CM documentation.

Content and presentation of evidence elements:

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

ACM_CAP.3.10C The CM system shall provide measures such that only authorized changes are made to the configuration items.

Evaluator action elements:

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

ACM_SCP.1 TOE CM coverage

Objectives

A CM system can control changes only to those items that have been placed under CM. (i.e., the configuration items identified in the configuration item list). Placing the TOE implementation and the evaluation evidence required by the other assurance components in the ST under CM provides assurance that they have been modified in a controlled manner with proper authorizations.

Dependencies:

ACM_CAP.3 Authorization controls

Developer action elements:

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

Content and presentation of evidence elements:

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

Evaluator action elements:

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

ADO_DEL.1 Delivery procedures

Dependencies:

No dependencies.

Developer action elements:

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

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

Content and presentation of evidence elements:

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

Evaluator action elements:

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

ADO_IGS.1 Installation, generation, and start-up procedures

Dependencies:

No dependencies.

Developer action elements:

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

Content and presentation of evidence elements:

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

Evaluator action elements:

ADO_IGS.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
ADO_IGS.1.2E	The evaluator shall determine that the installation, generation, and start-up procedures result in a secure configuration.

ADV_FSP.1 Informal functional specification

Dependencies:

ADV_RCR.1	Informal correspondence demonstration

Developer action elements:

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

Content and presentation of evidence elements:

ADV_FSP.1.1C	The functional specification shall describe the TSF and its external interfaces using an informal style.
ADV_FSP.1.2C	The functional specification shall be internally consistent.
ADV_FSP.1.3C	The functional specification shall describe the purpose and method of use of all external TSF interfaces, providing details of effects, exceptions and error messages, as appropriate.
ADV_FSP.1.4C	The functional specification shall completely represent the TSF.

Evaluator action elements:

ADV_FSP.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
ADV_FSP.1.2E	The evaluator shall determine that the functional specification is an accurate and complete instantiation of the TOE security functional

requirements.

ADV_HLD.2 Security enforcing high-level design

Dependencies:

ADV_FSP.1	Informal functional specification
ADV_RCR.1	Informal correspondence demonstration

Developer action elements:

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

Content and presentation of evidence elements:

ADV_HLD.2.1C	The presentation of the high-level design shall be informal.
ADV_HLD.2.2C	The high-level design shall be internally consistent.
ADV_HLD.2.3C	The high-level design shall describe the structure of the TSF in terms of subsystems.
ADV_HLD.2.4C	The high-level design shall describe the security functionality provided by each subsystem of the TSF.
ADV_HLD.2.5C	The high-level design shall identify any underlying hardware, firmware, and/or software required by the TSF with a presentation of the functions provided by the supporting protection mechanisms implemented in that hardware, firmware, or software.
ADV_HLD.2.6C	The high-level design shall identify all interfaces to the subsystems of the TSF.
ADV_HLD.2.7C	The high-level design shall identify which of the interfaces to the subsystems of the TSF are externally visible.
ADV_HLD.2.8C	The high-level design shall describe the purpose and method of use of all interfaces to the subsystems of the TSF, providing details of effects, exceptions and error messages, as appropriate.
ADV_HLD.2.9C	The high-level design shall describe the separation of the TOE into TSPenforcing and other subsystems.

Evaluator action elements:

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

ADV_HLD.2.2E The evaluator shall determine that the high-level design is an accurate and complete instantiation of the TOE security functional requirements.

ADV_RCR.1 Informal correspondence demonstration

Dependencies:

No dependencies.

Developer action elements:

ADV_RCR.1.1D The developer shall provide an analysis of correspondence between all adjacent pairs of TSF representations that are provided.

Content and presentation of evidence elements:

ADV_RCR.1.1C For each adjacent pair of provided TSF representations, the analysis shall demonstrate that all relevant security functionality of the more abstract TSF representation is correctly and completely refined in the less abstract TSF representation.

Evaluator action elements:

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

AGD_ADM.1 Administrator guidance

Dependencies:

ADV_FSP.1 Informal functional specification

Developer action elements:

AGD_ADM.1.1D The developer shall provide administrator guidance addressed to system administrative personnel.

Content and presentation of evidence elements:

AGD_ADM.1.1C The administrator guidance shall describe the administrative functions and interfaces available to the administrator of the TOE.

AGD_ADM.1.2C The administrator guidance shall describe how to administer the TOE in a secure manner.

AGD_ADM.1.3C The administrator guidance shall contain warnings about functions and privileges that should be controlled in a secure processing environment.

AGD_ADM.1.4C The administrator guidance shall describe all assumptions regarding user behaviour that are relevant to secure operation of the TOE.

AGD_ADM.1.5C The administrator guidance shall describe all security parameters under the control of the administrator, indicating secure values as appropriate.

AGD_ADM.1.6C The administrator guidance shall describe each type of security-relevant event relative to the administrative functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.

AGD_ADM.1.7C The administrator guidance shall be consistent with all other documentation supplied for evaluation.

AGD_ADM.1.8C The administrator guidance shall describe all security requirements for the IT environment that are relevant to the administrator.

Evaluator action elements:

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

AGD_USR.1 User guidance

Dependencies:

ADV_FSP.1 Informal functional specification

Developer action elements:

AGD_USR.1.1D The developer shall provide user guidance.

Content and presentation of evidence elements:

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

AGD_USR.1.2C The user guidance shall describe the use of user-accessible security functions provided by the TOE.

AGD_USR.1.3C The user guidance shall contain warnings about user-accessible functions and privileges that should be controlled in a secure processing environment.

AGD_USR.1.4C The user guidance shall clearly present all user responsibilities necessary for secure operation of the TOE, including those related to assumptions regarding user behaviour found in the statement of TOE security environment.

AGD_USR.1.5C The user guidance shall be consistent with all other documentation supplied for evaluation.

AGD_USR.1.6C The user guidance shall describe all security requirements for the IT environment that are relevant to the user.

Evaluator action elements:

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

ALC_DVS.1 Identification of security measures

Dependencies:

No dependencies.

Developer action elements:

ALC_DVS.1.1D The developer shall produce development security documentation.

Content and presentation of evidence elements:

ALC_DVS.1.1C The development security documentation shall describe all the physical, procedural, personnel, and other security measures that are necessary to protect the confidentiality and integrity of the TOE design and implementation in its development environment.

ALC_DVS.1.2C The development security documentation shall provide evidence that these security measures are followed during the development and maintenance of the TOE.

Evaluator action elements:

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

ALC_DVS.1.2E The evaluator shall confirm that the security measures are being applied.

ATE_COV.2 Analysis of coverage

Objectives

In this component, the objective is to establish that the TSF has been tested against its functional specification in a systematic manner. This is to be achieved through an examination of developer analysis of correspondence.

Application notes

The developer is required to demonstrate that the tests which have been identified include testing of all of the security functions as described in the functional specification. The analysis should not only show the correspondence between tests and security functions, but should provide also sufficient information for the evaluator to determine how the functions have been exercised. This information can be used in planning for additional evaluator tests. Although at this level the developer has to demonstrate that each of the functions within the functional specification has been tested, the amount of testing of each function need not be exhaustive.

Dependencies:

ADV_FSP.1	Informal functional specification
ATE_FUN.1	Functional testing

Developer action elements:

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

Content and presentation of evidence elements:

ATE_COV.2.1C	The analysis of the test coverage shall demonstrate the correspondence between the tests identified in the test documentation and the TSF as described in the functional specification.
ATE_COV.2.2C	The analysis of the test coverage shall demonstrate that the correspondence between the TSF as described in the functional specification and the tests identified in the test documentation is complete.

Evaluator action elements:

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

ATE_DPT.1 Testing: high-level design

Objectives

The subsystems of a TSF provide a high-level description of the internal workings of the TSF. Testing at the level of the subsystems, in order to demonstrate the presence of any flaws, provides assurance that the TSF subsystems have been correctly realized.

Application notes

The developer is expected to describe the testing of the high-level design of the TSF in terms of “subsystems”. The term “subsystem” is used to express the notion of decomposing the TSF into a relatively small number of parts.

Dependencies:

ADV_HLD.1	Descriptive high-level design
ATE_FUN.1	Functional testing

Developer action elements:

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

Content and presentation of evidence elements:

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

Evaluator action elements:

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

ATE_FUN.1 Functional testing

Objectives

The objective is for the developer to demonstrate that all security functions perform as specified. The developer is required to perform testing and to provide test documentation.

Dependencies:

No dependencies.

Developer action elements:

ATE_FUN.1.1D	The developer shall test the TSF and document the results.
ATE_FUN.1.2D	The developer shall provide test documentation.

Content and presentation of evidence elements:

ATE_FUN.1.1C	The test documentation shall consist of test plans, test procedure

descriptions, expected test results and actual test results.

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

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

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

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

Evaluator action elements:

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

ATE_IND.2 Independent testing - sample

Objectives

The objective is to demonstrate that the security functions perform as specified. Evaluator testing includes selecting and repeating a sample of the developer tests.

Application notes

The intent is that the developer should provide the evaluator with materials necessary for the efficient reproduction of developer tests. This may include such things as machine-readable test documentation, test programs, etc.

This component contains a requirement that the evaluator has available test results from the developer to supplement the program of testing. The evaluator will repeat a sample of the developer's tests to gain confidence in the results obtained. Having established such confidence the evaluator will build upon the developer's testing by conducting additional tests that exercise the TOE in a different manner. By using a platform of validated developer test results the evaluator is able to gain confidence that the TOE operates correctly in a wider range of conditions than would be possible purely using the developer's own efforts, given a fixed level of resource. Having gained confidence that the developer has tested the TOE, the evaluator will also have more freedom, where appropriate, to concentrate testing in areas where examination of documentation or specialist knowledge has raised particular concerns.

Dependencies:

ADV_FSP.1	Informal functional specification
AGD_USR.1	User guidance
ATE_FUN.1	Functional testing

Developer action elements:

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

Content and presentation of evidence elements:

ATE_IND.2.1C	The TOE shall be suitable for testing.
ATE_IND.2.2C	The developer shall provide an equivalent set of resources to those that were used in the developer's functional testing of the TSF.

Evaluator action elements:

ATE_IND.2.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
ATE_IND.2.2E	The evaluator shall test a subset of the TSF as appropriate to confirm that the TOE operates as specified.
ATE_IND.2.3E	The evaluator shall execute a sample of tests in the test documentation to verify the developer test results.

AVA_MSU.1 Examination of guidance

Objectives

The objective is to ensure that misleading, unreasonable and conflicting guidance is absent from the guidance documentation, and that secure procedures for all modes of operation have been addressed. Insecure states should be easy to detect.

Dependencies:

ADO_IGS.1	Installation, generation, and start-up procedures

ADV_FSP.1	Informal functional specification
AGD_USR.1	User guidance

Developer action elements:

AVA_MSU.1.1D	The developer shall provide guidance documentation.

Content and presentation of evidence elements:

AVA_MSU.1.1C	The guidance documentation shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences and implications for maintaining secure operation.
AVA_MSU.1.2C	The guidance documentation shall be complete, clear, consistent and reasonable.
AVA_MSU.1.3C	The guidance documentation shall list all assumptions about the intended environment.
AVA_MSU.1.4C	The guidance documentation shall list all requirements for external security measures (including external procedural, physical and personnel controls).

Evaluator action elements:

AVA_MSU.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
AVA_MSU.1.2E	The evaluator shall repeat all configuration and installation procedures to confirm that the TOE can be configured and used securely using only the supplied guidance documentation.
AVA_MSU.1.3E	The evaluator shall determine that the use of the guidance documentation allows all insecure states to be detected.

AVA_SOF.1 Strength of TOE security function evaluation

Dependencies:

ADV_FSP.1	Informal functional specification
ADV_HLD.1	Descriptive high-level design

Developer action elements:

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

Content and presentation of evidence elements:

AVA_SOF.1.1C	For each mechanism with a strength of TOE security function claim the strength of TOE security function analysis shall show that it meets or exceeds the minimum strength level defined in the PP/ST.
AVA_SOF.1.2C	For each mechanism with a specific strength of TOE security function claim the strength of TOE security function analysis shall show that it meets or exceeds the specific strength of function metric defined in the PP/ST.

Evaluator action elements:

AVA_SOF.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
AVA_SOF.1.2E	The evaluator shall confirm that the strength claims are correct.

AVA_VLA.1 Developer vulnerability analysis

Objectives

A vulnerability analysis is performed by the developer to ascertain the presence of obvious security vulnerabilities, and to confirm that they cannot be exploited in the intended environment for the TOE.

Application notes

The evaluator should consider performing additional tests as a result of potential exploitable vulnerabilities identified during other parts of the evaluation.

Dependencies:

ADV_FSP.1	Informal functional specification
ADV_HLD.1	Descriptive high-level design
AGD_USR.1	User guidance

Developer action elements:

AVA_VLA.1.1D	The developer shall perform a vulnerability analysis.
AVA_VLA.1.2D	The developer shall provide vulnerability analysis documentation.

Content and presentation of evidence elements:

AVA_VLA.1.1C	The vulnerability analysis documentation shall describe the analysis of the TOE deliverables performed to search for obvious ways in which a user can violate the TSP.
AVA_VLA.1.2C	The vulnerability analysis documentation shall describe the disposition of obvious vulnerabilities.
AVA_VLA.1.3C	The vulnerability analysis documentation shall show, for all identified vulnerabilities, that the vulnerability cannot be exploited in the intended environment for the TOE.

Evaluator action elements:

AVA_VLA.1.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
AVA_VLA.1.2E	The evaluator shall conduct penetration testing, building on the developer vulnerability analysis, to ensure obvious vulnerabilities have been addressed.

6 TOE SUMMARY SPECIFICATION

This section provides a description of the security functions and assurance measures of the TOE that meet the TOE security requirements.

A typical attacker in the intended environment for the TOE is assumed to have a low level of sophistication, but may have knowledge of vulnerabilities and access to attack methods that are in the public domain. The purpose of the attacks could be (1) to gain access to the resources of the TOE, (2) to gain access to the resources of the client systems protected by the TOE, and/or (3) to prevent the successful remediation of client systems and thus leave these systems in a vulnerable state. Therefore, the attack potential which is applicable for AVA_SOF.1 calculations is LOW. Any residual vulnerabilities may only be exploited by an attacker of moderate or high attack potential. The strength of function claim is therefore SOF-BASIC. This claim applies to the security function F.IAUSER, F.IACONSOLE, F.IACLIENT and F.IAREMSRV.

6.1 TOE SECURITY FUNCTIONS

A description of each of the TOE security functions follows.

F.AGGVADATA — Aggregate Scanner Data

The TOE has the capability of merging vulnerability scanner information from the third party vulnerability scanners for a client machine into a single consistent vulnerability assessment for that machine.

F.APPPROF — Approve Profile

The TOE provides the capability for a suitably authorized user to approve a remediation profile. Once approved the remediation profile shall be automatically invoked by each client machine in the group to which the profile applies at the next scheduled remediation interval.

F.AUDIT — Audit Remediation Activity

The TOE maintains an audit trail of remediation activity performed by each Hercules® AVR server. The Hercules® AVR server components and windows client systems create events in the Windows event logs which include stop, start, successful actions and failed actions. These events are created on the Hercules® AVR server and the target windows machine which is being remediated. The Hercules® AVR server is capable of generating audit events associated with the Windows Event Viewer application,

	security and system categories.
F.DISPCLIENT	Display Network Client Systems The TOE has the capability of displaying via a graphical user interface a list of devices connected to a Hercules® AVR Server.
F.DISPCLIENTSTATUS	Display Network Client Status The TOE has the capability of displaying via a graphical user interface the operational status of each client machine.
F.DISPPROF	Display Profiles The TOE has the capability of displaying via a graphical user interface, the list of vulnerabilities which will be remediated by the Hercules® AVR Server for a client machine or a group of client machines.
F.DISPREMSTATUS	Display Remediation Status The TOE has the capability of displaying via a graphical user interface the remediation status of each client machine of each Hercules® AVR Server.
F.DISPSIG	Display Remediation Signatures The TOE has the capability of displaying via a graphical user interface, the steps required to remediate a specific vulnerability on a client machine.
F.DISPVADATA	Display Scanner Data The TOE has the capability of displaying imported scanner information.
F.DISPVULN	Display Vulnerabilities The TOE has the capability of displaying graphically the vulnerabilities of machines on a network. It shall be possible to list all of the vulnerabilities reported for each and all machines on the network, or to display a list of machines which are susceptible to a specific vulnerability.
F.DOMAINSEP	Domain Separation The TOE maintains a security domain for its own execution which protects the TOE from interference and tampering by

	untrusted subjects. The TOE enforces the separation of the security domains of the client systems which are being remediated. The operating systems (Windows and Unix) provide this security domain in order to protect the TOE and provide process isolation. The TSF ensures that TSP enforcement functions are invoked and succeed before each function within the TSC is allowed to proceed. The TSF enforces separation between the security domains of subjects in the TSC.
F.ENCRYPT	Encrypt Data The TOE has the capability of encrypting data which is transferred between the physically separate elements of the TOE. The user can configure the Hercules® AVR Administrator Console to use HTTPS communication to the Hercules® AVR Remediation Server. The user can configure the Hercules® AVR clients to use HTTPS communication to the Hercules® AVR Remediation Server. All Hercules® AVR Remediation Server to Unix client communications for client management actions will be via SSH. All communication between a Hercules® AVR Remediation Server and the Citadel Vflash server will be via HTTPS. If a patch vendor's site supports HTTPS, the downloading of patches will use HTTPS.
F.IACLIENT	Identify and Authenticate Clients Each Hercules® AVR Remediation Server has the capability to identify and authenticate each client machine for which it will issue a remediation profile. The client machines can be configured for HTTPS authentication with the Hercules® AVR server using a server certificate. In the evaluated configuration, the clients machines shall be configured with a client certificate for mutual authentication with the Hercules® AVR server.
F.IAREMSVR	Identify and Authenticate Remediation Server The Hercules® AVR Administrator Console has the capability to identify and authenticate each Hercules® AVR Remediation Server through the use of a certificate installed on the server. The Hercules® AVR Server has the ability to authenticate to the Windows Domain Controller using a Domain Administrator account with an internally generated,

	random password.
F.IAUSER	Identify and Authenticate Users The Hercules® AVR Administrator Console has the capability to identify and authenticate users of the console. The Hercules® AVR Administrator Console executes using a Windows administrator account which is recognized by the machine hosting the Hercules® AVR server.
F.IMPREMDATA	Import Remediation Data The TOE has the capability to import specific remediation information for reported vulnerabilities.
F.IMPVADATA	Import Scanner Data The TOE has the capability of importing vulnerability scanner information from the following third party vulnerability scanners: 1. Foundstone® FoundScan Engine™ 2. Harris STAT® Scanner 3. ISS Internet Scanner® 4. ISS System Scanner® 5. Microsoft® MBSA 6. Nessus Scanner 7. Qualys QualysGuard™ Scanner 8. Retina® Digital Security Scanner 9. VIGILANTe SecureScan™

F.MANAGEDATA	Manage Scanner and Remediation Data The TOE provides the user with an interface from which it is possible to manage the vulnerability scanner information and the vulnerability remediation information. A user may view a remediation profile for a device in order to determine which vulnerabilities and associated remedies will be applied to a device when it is remediated.
F.MANAGEPROF	Manage Profiles The TOE provides the capability for a suitably authorized user to manage remediation profiles. Machines may be added to or removed from the group to which the profile applies. Specific vulnerabilities may be added to or removed from the remediation profile.
F.MANAGEROLES	Manage Roles The TOE provides the capability for a suitably authorized user to create and manage custom roles for the TOE. Once created, individual users and groups of users may be assigned to the role. Privileges to use specific functions of the TOE such as creating custom remediation remedies and user defined vulnerabilities may also be assigned to the role.
F.PUSHREM	Push Remediation Data The Hercules® AVR Server provides remediation data in the form of a remediation profile to client machines.
F.REMCLIENT	Remediate Client System The TOE provides the capability to automatically remediate specific vulnerabilities on client machines.
F.REPREMSTATUS	Report Remediation Status The TOE has the capability of producing reports describing the remediation status of each client machine of each Hercules® AVR Server. The user can select reports which show the details and summaries of; remediation sessions, import sessions, devices, groups, vulnerabilities, policies and remedies.
F.ROLLBACK	Rollback Remediations The TOE has the capability to systematically rollback the

	last remediation session performed on a Windows® client machine.
F.RVM	Reference Monitor This TOE security function is supported by the operating systems (Windows and Unix) to provide reference mediation (e.g., when a user process requires access to a resource its requests a handle/token for the resource from the operating system).
F.SCHEDREM	Schedule Remediations The TOE provides the capability to schedule remediation activity for single client machines or groups of client machines.

6.2 ASSURANCE MEASURES

A description of each of the TOE assurance measures follows.

M.AUTH	The TOE includes documentation which describes the authorization controls used by the developer to ensure that only authorized modifications may be made to the TOE.
M.CONFIG	The TOE includes a configuration item list which identifies those items of the TOE which are subject to configuration control by the developer.
M.DELIVER	The TOE includes documentation describing the secure delivery of the TOE.
M.DESIGN	The TOE includes design documentation which at a minimum consists of an informal functional specification, an informal high level design and an informal correspondence demonstration between the TOE Summary Specification, the Functional Specification and the High Level Design.
M.DEVELOP	The TOE includes documentation which describes the development security measures.
M.DOCS	The TOE includes user and administrator guidance documentation in the form of a User's Guide and an Installation Guide as well as an on-line, help file, accessible from the TOE HMI.
M.ID	The TOE incorporates a unique version identifier that can be displayed to the user.
M.SETUP	The TOE includes an automated installation and set-up program compatible with the TOE operating system. The installation process includes sufficient instructions to clearly document the installation process. The default installation results in the secure installation and start-

	up of the TOE.
M.TEST	A suitably configured TOE has been evaluated in a controlled networked environment to confirm that TOE functionality operates as specified, and that the product can remediate a representative set of well-known vulnerabilities from each of the vulnerability classes claimed by the developer. TOE functionality has also been evaluated in a real-world environment, using a representative set of network systems configured with known vulnerabilities. The TOE includes developer test documentation which consists of test plans, test procedure descriptions, expected test results and actual test results. The test documentation is sufficient to determine that the developer has systematically tested the TOE against both the functional specification and the high level design.
M.VULNER	The TOE includes vulnerability documentation which describes the strength of function analysis along with an analysis of obvious vulnerabilities in the TOE.

7 PROTECTION PROFILE CLAIMS

This ST does not make compliance claims with respect to any Protection Profiles.

8 RATIONALE

8.1 SECURITY OBJECTIVES RATIONALE

Table 3 provides a bi-directional mapping of Security Objectives to Threats and Assumptions. It is followed by a discussion of how each Threat or Assumption is addressed by the corresponding Security Objective(s).

	O.CLIENTAUTH	O.CLIENTPROT	O.CLIENTREM	O.HMI	O.NETATK	O.REMDATA	O.SCANDATA	OE.AUTHUSER	OE.BACKUP	OE.GOODOS	OE.GOODUSER	OE.GU I DANCE	OE.SECURECOM
A.BACKUP									X				
A.CMS													X
A.CONFIG												X	
A.GOODOS										X			
A.KNOWLEDGE											X		
A.NOEVIL											X		
A.PHYSICAL								X					
A.TOEUSER											X		
T.BADDATA						X							
T.CLIENT	X	X											
T.CONSOLE				X	X								
T.EXPLOIT			X										
T.NETEXPLOIT			X										
T.REMSERVER					X								
T.SNIFF					X								
T.SNIFFSCAN							X						
T.SPOOF	X												
T.SPOOFCLIENT	X												
T.SPOOFSCAN							X						

Table 3 Mapping of Security Objectives to Threats and Assumptions

A.BACKUP *The organization operating the TOE has good backup and recovery procedures which are followed; allowing the TOE to be recovered to*

	a secure configuration after a hardware failure. The OE.BACKUP objective details the need for good backup and recovery procedures.
A.CMS	*Windows client machines which will be remediated using Client Management Services (CMS) are assumed to reside on a protected network.* The OE.SECURECOM objective ensures that communications between the Hercules Server and Windows client machines using CMS are protected.
A.CONFIG	*The servers running the Remediation Server and the Administrator Console have been configured securely as described in the Guidance documents and are maintained in that secure configuration. In particular:* *a. They are configured with the minimal operating system features installed and / or enabled to permit operation of the TOE.* *b. They are configured with minimal system privileges.* *c. They are configured with user accounts for authorized system administrators only and do not provide any end user accounts.* The OE.GUIDANCE objective ensures that the TOE will be configured securely.
A.GOODOS	*The Operating System of the client machines has been configured in accordance with the Hercules® AVR Security Configuration Guide and therefore may be trusted to function correctly for those OS functions required by the TOE component that is installed on the client machine.* The OE.GOODOS objective ensures that those functions of the operating system required by the TOE function correctly.
A.KNOWLEDGE	*TOE Users have knowledge of the Windows® 2000/XP/2003 operating system, networking technology and general IT security practices.* The OE.GOODUSER objective notes that TOE Users must be knowledgeable.
A.NOEVIL	*TOE Users are non hostile and follow all guidance documents.* The OE.GOODUSER objective notes that TOE Users must be non malicious.
A.PHYSICAL	*The Server and Administrator elements of the TOE are physically secure and only authorized personnel have physical access to these elements of the TOE.* The OE.AUTHUSER objective notes that only authorized personnel are permitted physical access to the TOE.

A.TOEUSER	*There is only one category of TOE user. All authorized TOE users have full access to all of the TOE's functions and for this reason there is no distinction between TOE users and TOE administrators. For the remainder of this document the phrase 'TOE User' shall be employed.* The OE.GOODUSER objective describes the characteristics of the TOE Users and notes that these users must be authorized system administrators.
T.BADDATA	*A network attacker may attempt to provide the Remediation Server with erroneous remediation information in an attempt to compromise the Client systems.* The O.REMDATA objective ensures that the remediation data used by the TOE is accurate and secure.
T.CLIENT	*An unauthorized person may have administrator / root control of one of the client systems and may use that control to attempt to compromise the Remediation Server.* The O.CLIENTAUTH and O.CLIENTPROT objectives ensure that the TOE is protected against attacks by the client systems.
T.CONSOLE	*A network attacker may attempt to gain control of the TOE through the Hercules® AVR Administration Console.* The O.HMI and O.NETATK objectives ensure that the Administration Console is secure.
T.EXPLOIT	*A network attacker may attempt to exploit vulnerabilities on a Client system protected by the TOE in order to gain unauthorized access to the resources of the client system.* The O.CLIENTREM objective ensures that the TOE provides effective remediation to client systems in order to remove or mitigate identified vulnerabilities.
T.NETEXPLOIT	*A network attacker may attempt to exploit vulnerabilities on a Client system protected by the TOE in an attempt to compromise other network resources.* The O.CLIENTREM objective ensures that the TOE provides effective remediation to client systems in order to remove or mitigate identified vulnerabilities.
T.REMSERVER	*A network attacker may attempt to gain control of the Hercules® AVR Remediation Server* The O.NETATK objective ensures that the Remediation Server is secure.
T.SNIFF	*A network attacker may monitor communications between the Remediation Server and the Client systems and use the information gained to compromise the Remediation Server and / or a Client*

	system.
	The O.NETATK objective ensures that the information passing between the distributed parts of the TOE is secure.
T.SNIFFSCAN	*A network attacker may monitor communications between the Remediation Server and a vulnerability scanner to learn vulnerabilities of client systems.*
	The O.SCANDATA objective ensures that the scanner data used by the TOE is accurate and secure.
T.SPOOF	*A network attacker may attempt to imitate the Remediation Server and provide erroneous remediation information to a client system in order to compromise the client.*
	The O.CLIENTAUTH objective ensures that it is not possible to imitate the Remediation server.
T.SPOOFCLIENT	*A network attacker may attempt to imitate a client system in order to gain information about the vulnerabilities of the client system.*
	The O.CLIENTAUTH objective ensures that it is not possible for an attacker to imitate a client system.
T.SPOOFSCAN	*A network attacker may attempt to provide the Remediation Server with erroneous vulnerability assessment information in an attempt to prevent the remediation of vulnerable network systems.*
	The O.SCANDATA objective ensures that the scanner data used by the TOE is accurate and secure

8.2 SECURITY REQUIREMENTS RATIONALE

Table 4 provides a bi-directional mapping of Security Functional Requirements to Security Objectives, and is followed by a discussion of how each Security Objective is addressed by the corresponding Security Functional Requirements.

	O.CLIENTAUTH	O.CLIENTPROT	O.CLIENTREM	O.HMI	O.NETATK	O.REMDATA	O.SCANDATA
FAU_ GEN.1		X	X	X	X	X	X
FAU_ SAR.1				X			
FAU_ SEL.1				X			
FDP_ IFC.1	X	X	X			X	

	O.CLIENTAUTH	O.CLIENTPROT	O.CLIENTREM	O.HMI	O.NETATK	O.REMDATA	O.SCANDATA
FDP_ IFF.1	X	X	X			X	
FDP_ ITC.1						X	X
FDP_ ITT.1			X				
FDP_ROL.1			X				
FIA_AFL.1				X			
FIA_SOS.1				X			
FIA_UAU.2	X	X		X	X	X	
FIA_UID.2	X	X		X	X	X	
FMT_MSA.1	X		X	X			
FMT_ MSA.3	X		X	X			
FMT_MTD.1							X
FMT_SMF.1	X		X	X			
FMT_SMR.1				X			
FPT_ITT.1					X		
FPT_RVM.1			X				
FPT_SEP.1			X				
FPT_ STM.1				X			

Table 4 Mapping of Security Functional Requirements to TOE Security Objectives

O.CLIENTAUTH *The TOE must provide a mechanism for a two way authentication between client systems and the Remediation Server.*

The SERVER_SFP information flow control security functional policy and associated management functions (FDP_IFC.1, FDP_IFF.1, FMT_MSA.1, FMT_MSA.3, FMT_SMF.1) control the flow of information between the Hercules® AVR Server and the client systems. In addition, the identification and authentication functional requirements (FIA_UAU.2 and FIA_UID.2) ensure that the identification and authentication activities complete successfully before information is transferred.

O.CLIENTPROT *The TOE must protect itself against attacks initiated by client*

	systems. The TOE will only respond to requests for remediations which are received from identified and authorized client machines (FIA_UAU.2, FIA_UID.2). The TOE also enforces the SERVER_SFP information flow control security functional policy to limit its exposure to attacks by client machines and to ensure that only the correct remediation profiles are provided to client machines (FDP_IFC.1, FDP_IFF.1). The TOE also maintains an audit trail of remediation requests which may help to identify an attack from a client machine (FAU_GEN.1).
O.CLIENTREM	*The TOE must provide effective remediation of known and reported vulnerabilities for client systems.* The TOE obtains its vulnerability and remediation data from trusted external sources using the IMPORT_SFP information flow control security function policy to govern the data import process (FDP_IFC.1, FDP_IFF.1). The TOE protects its data from unauthorized modifications or corruption, both internally (FMT_MSA.1, FMT_MSA.3, FPT_RVM.1, FPT_SEP.1) and during transmission to the client systems (FDP_ITT.1). The TOE enforces the SERVER_SFP information flow control security functional policy when providing specific remediation data to authorized client systems (FDP_IFC.1, FDP_IFF.1). The TOE permits authorized users to configure the list of client systems and vulnerabilities which will be remediated (FMT_SMF.1). Under specific circumstances the TOE is capable of rolling back remediations (FDP_ROL.1). Finally, the TOE maintains a comprehensive audit trail of its actions (FAU_GEN.1).
O.HMI	*The TOE must provide a controlled interface to its functionality such that only authorized TOE users are able to access the interface.* The TOE HMI is provided by the Hercules® AVR Administrator Console. This component of the TOE is only accessible to authorized administrative users (FIA_AFL.1, FIA_UAU.2, FIA_UID.2, FMT_SMR.1). Authorized users of the Hercules® AVR Administrator may control all of the security functions of the TOE, including setting security attributes and importing vulnerability scan and remediation data (FIA_SOS.1, FMT_MSA.1, FMT_MSA.3, FMT_SMF.1). Actions performed by authorized users are subject to auditing (FAU_GEN.1, FAU_SAR.1, FAU_SEL.1, FPT_STM.1).
O.NETATK	*The TOE must protect itself against network attackers.* The TOE protects itself against network attackers through its identification and authentication functions (FIA_UAU.2, FIA_UID.2). The TOE also protects its data from disclosure and modification while transmitting this data to the client systems (FPT_ITT.1). The collection of audit data (FAU_GEN.1) ensures

that attacks of this type will be detected.

O.REMDATA *The TOE must ensure that its remediation data is obtained from trusted sources and must provide a mechanism to ensure the integrity of this data.*

After initial installation, the TOE obtains its remediation data updates either from manual entry by an authorized user or by remote download from the Hercules® AVR VFlash server. Since all Hercules® AVR users are subject to the I&A mechanisms of the product (FIA_UAU.2, FIA_UID.2) it follows that only authorized and identified users may manually create remediation data. The product also enforces the IMPORT_SFP information flow security functional policy (FDP_IFC.1, FDP_IFF.1, FDP_ITC.1) when importing remediation data from the V Flash server. This ensures that the remediation data is obtained from a trusted source. The TOE maintains an audit record of import sessions (FAU_GEN.1) so that it is possible to confirm that the product has current, accurate and valid remediation data.

O.SCANDATA *The TOE must ensure that its scanner data is obtained from trusted sources and must provide a mechanism to ensure the integrity of this data.*

The TOE enforces the IMPORT_SFP information flow control security functional policy (FDP_ITC.1) to ensure that only trusted scanner data is imported by the TOE. Once under the control of the TOE, the scanner data may only be accessed by authorized TOE users (FMT_MTD.1). This ensures the integrity of the data. The audit trail records the details of scanner data import sessions (FAU_GEN.1).

8.3 SECURITY FUNCTIONAL REQUIREMENT DEPENDENCIES

Table 5 identifies the TOE Security Functional Requirements and their associated dependencies. It also indicates whether the TOE explicitly addresses each dependency. Notes are provided for those cases where the dependencies are satisfied by components which are hierarchical to the specified dependency.

Security Functional Requirement	**Dependencies**	**Dependency Satisfied**	**Notes**
FAU_ GEN.1	FPT_STM.1	Yes	
FAU_SAR.1	FAU_GEN.1	Yes	
FAU_ SEL.1	FAU_GEN.1	Yes	

Security Functional Requirement	Dependencies	Dependency Satisfied	Notes
	FMT_MTD.1	Yes	
FDP_ IFC.1	FDP_IFF.1	Yes	
FDP_ IFF.1	FDP_IFC.1	Yes	
	FMT_MSA.3	Yes	
FDP_ ITC.1	FDP_IFC.1	Yes	
	FMT_MSA.3	Yes	
FDP_ITT.1	FDP_IFC.1	Yes	
FDP_ROL.1	FDP_IFC.1	Yes	
FIA_AFL.1	FIA_UAU.1	Yes	FIA_UAU.2 is specified as a security functional requirement and FIA_UAU.2 is hierarchical to FIA_UAU.1
FIA_SOS.1	None	N/A	
FIA_UAU.2	FIA_UID.1	Yes	FIA_UID.2 is specified as a security functional requirement and FIA_UID.2 is hierarchical to FIA_UID.1.
FIA_UID.2	None	N/A	
FMT_MSA.1	FDP_IFC.1	Yes	
	FMT_SMF.1	Yes	
	FMT_SMR.1	Yes	
FMT_ MSA.3	FMT_MSA.1	Yes	
	FMT_SMR.1	Yes	
FMT_MTD.1	FMT_SMF.1	Yes	
	FMT_SMR.1	Yes	
FMT_SMF.1	None	N/A	
FMT_SMR.1	FIA_UID.1	Yes	FIA_UID.2 is specified as a security functional requirement and FIA_UID.2 is hierarchical to FIA_UID.1.
FPT_ITT.1	None	N/A	
FPT_RVM.1	None	N/A	
FPT_SEP.1	None	N/A	

Security Functional Requirement	Dependencies	Dependency Satisfied	Notes
FPT_STM.1	None	N/A	

Table 5 Security Functional Requirement Dependencies

8.4 SECURITY ASSURANCE REQUIREMENT DEPENDENCIES

Security Functional Requirement	Dependencies	Dependency Satisfied	Notes
ACM_CAP.3	ACM_DVS.1	Yes	
ACM_SCP.1	ACM_CAP.3	Yes	
ADO_DEL.1	None	N/A	
ADO_IGS.1	None	N/A	
ADV_FSP.1	ADV_RCR.1	Yes	
ADV_HLD.2	ADV_FSP.1	Yes	
	ADV_RCR.1	Yes	
ADV_RCR.1	None	N/A	
AGD_ADM.1	ADV_FSP.1	Yes	
AGD_USR.1	ADV_FSP.1	Yes	
ALC_DVS.1	None	N/A	
ATE_COV.2	ADV_FSP.1	Yes	
	ATE_FUN.1	Yes	
ATE_DPT.1	ADV_HLD.1	Yes	ADV_HLD.2 is specified as a security assurance requirement and ADV_HLD.2 is hierarchical to ADV_HLD.1.
	ATE_FUN.1	Yes	
ATE_FUN.1	None	N/A	
ATE_IND.2	ADV_FSP.1	Yes	
	AGD_USR.1	Yes	
	ATE_FUN.1	Yes	

Security Functional Requirement	Dependencies	Dependency Satisfied	Notes
AVA_MSU.1	ADO_IGS.1	Yes	
	ADV_FSP.1	Yes	
	AGD_USR.1	Yes	
AVA_SOF.1	ADV_FSP.1	Yes	
	ADV_HLD.1	Yes	ADV_HLD.2 is specified as a security assurance requirement and ADV_HLD.2 is hierarchical to ADV_HLD.1.
AVA_VLA.1	ADV_FSP.1	Yes	
	ADV_HLD.1	Yes	ADV_HLD.2 is specified as a security assurance requirement and ADV_HLD.2 is hierarchical to ADV_HLD.1.
	AGD_USR.1	Yes	

Table 6 Security Assurance Requirement Dependencies

8.5 TOE SUMMARY SPECIFICATION RATIONALE

Table 7 provides a bi-directional mapping of Security Functions to Security Functional Requirements, and is followed by a discussion of how each Security Functional Requirement is addressed by the corresponding Security Function.

	FAU_GEN.1	FAU_SAR.1	FAU_SEL.1	FDP_IFC.1	FDP_IFF.1	FDP_ITC.1	FDP_ITT.1	FDP_ROL.1	FIA_AFL.1	FIA_SOS.1(1)	FIA_SOS.1(2)	FIA_UAU.2	FIA_UID.2	FMT_MSA.1	FMT_MSA.3	FMT_MTD.1	FMT_SMF.1	FMT_SMR.1	FPT_ITT.1	FPT_RVM.1	FPT_SEP.1	FPT_STM.1
F.AGGVADATA														X		X						
F.APPPROF																X	X					
F.AUDIT	X	X	X																			X
F.DISPCLIENT																X						
F.DISPCLIENTSTATUS																X						
F.DISPPROF																X						
F.DISPREMSTATUS																X						

	FAU_GEN.1	FAU_SAR.1	FAU_SEL.1	FDP_IFC.1	FDP_IFF.1	FDP_ITC.1	FDP_ITT.1	FDP_ROL.1	FIA_AFL.1	FIA_SOS.1(1)	FIA_SOS.1(2)	FIA_UAU.2	FIA_UID.2	FMT_MSA.1	FMT_MSA.3	FMT_MTD.1	FMT_SMF.1	FMT_SMR.1	FPT_ITT.1	FPT_RVM.1	FPT_SEP.1	FPT_STM.1
F.DISPSIG																X						
F.DISPVADATA														X		X						
F.DISPVULN																X						
F.DOMAINSEP																					X	
F.ENCRYPT							X												X			
F.IACLIENT				X	X		X		X		X	X	X		X				X			
F.IAREMSVR							X		X			X	X									
F.IAUSER									X	X		X	X	X	X	X		X				
F.IMPREMDATA				X	X	X								X								
F.IMPVADATA				X	X	X								X								
F.MANAGEDATA						X								X			X					
F.MANAGEPROF																X	X					
F.MANAGEROLES																		X				
F.PUSHREM				X	X		X										X		X			
F.REMCLIENT				X	X		X										X					
F.REPREMSTATUS		X																				
F.ROLLBACK								X									X					
F.RVM																				X		
F.SCHEDREM																X	X					

Table 7 Mapping of Security Functions to Security Functional Requirements

FAU_GEN.1 *Audit data generation*

The audit function of the TOE collects (F.AUDIT) and stores audit data for actions which are specific to the TOE (scanner data import, remediation data import, client remediations). In addition, the operating system audit trail retains audit records related to the identification and authorization of users, the start up and shut down of the TOE and the start up and shut down of the OS audit mechanism.

FAU_SAR.1 *Audit review*

The TOE includes a comprehensive HMI (Hercules® AVR Administrator Console) with extensive display and reporting features (F.REPREMSTATUS) which permit all authorized users with the ability to review, scan, analyze and interpret the audit trail recorded by the TOE (F.AUDIT).

FAU_SEL.1 *Selective Audit*

The TOE HMI (Hercules® AVR Administrator Console) provides authorized users with the ability to view audit information based both upon specific vulnerabilities or upon specific client machines or group of machines (F.AUDIT).

FDP_IFC.1 *Subset information flow control*

Each Hercules® AVR Server enforces the SERVER_SFP information flow control security functional policy which dictates that the server must identify and authenticate a client machine (F.IACLIENT) before accepting a request for remediation data from that client and providing the remediation profile (F.PUSHREM) which is used to remediate the client (F.REMCLIENT)

Each Hercules® AVR Server also enforces the IMPORT_SFP information flow control security functional policy when importing both vulnerability scan data (F.IMPVADATA) and vulnerability remediation data (F.IMPREMDATA).

FDP_IFF.1 *Simple security attributes*

The TOE uses the SERVER_SFP information flow control security functional policy to govern the exchange of data between a Hercules® AVR Server and one of its client systems. This policy states that the server must identify and authenticate the client (F.IACLIENT) before providing the client with the remediation information (F.PUSHREM) necessary to remediate the vulnerabilities on the client system (F.REMCLIENT).

The TOE uses the IMPORT_SFP information flow control security functional policy to govern the import of vulnerability scan information (F.IMPVADATA) and vulnerability remediation data (F.IMPREMDATA) from trusted external sources.

FDP_ITC.1 *Import of user data without security attribute*

When importing vulnerability scan data (F.IMPVADATA) or vulnerability remediation data (F.IMPREMDATA) from trusted external sources, the TOE ignores any security attributes associated with the external data and instead applies the properties specified by the authorized TOE user (F.MANAGEDATA) to the imported data.

FDP_ITT.1 *Basic internal transfer protection*

The Hercules® AVR Server enforces the SERVER_SFP information flow control security functional policy (F.IACLIENT, F.PUSHREM, F.REMCLIENT) to protect its remediation data from disclosure or modification while being transmitted from the server to a client system. The Hercules® AVR server has the ability to encrypt data transferred to a client system (F.ENCRYPT) using SSL for Windows® clients and OpenSSH for Unix clients.

FDP_ROL.1 *Basic Rollback*

The TOE allows the rollback (F.ROLLBACK) of specific automatic vulnerability remediations under specified circumstances.

FIA_AFL.1 *Authentication failure handling*

The TOE relies upon the identification and authentication mechanisms of the underlying operating system in order to identify and authenticate individual users of the TOE HMI (F.IAUSER). In addition, the Hercules® AVR Administrator Console and Hercules® AVR Server perform mutual identification and authentication before exchanging information (F.IAREMSVR), and a Hercules® AVR server identifies and authenticates each client system (F.IACLIENT) before providing any remediation data to that client.

FIA_SOS.1 *Verification of Secrets*

The TOE relies upon the underlying operating system for the entry and management of passwords for authorized users. The TOE guidance documents provide instructions concerning the minimum standards required for secure passwords (F.IAUSER). Additionally, the TOE uses functions available in the operating system to randomly generate a password which is used to authenticate to the Windows domain controller.

FIA_UAU.2 *User authentication before any action*

All of the identification and authentication mechanisms used by the TOE (F.IAUSER, F.IACLIENT, F.IAREMSVR), require complete and successful authentication before allowing any action to be performed.

FIA_UID.2 *User identification before any action*

All of the identification and authentication mechanisms used by the TOE (F.IAUSER, F.IACLIENT, F.IAREMSVR), require successful identification either of the individual user or the requesting system, before allowing any action to be performed.

FMT_MSA.1 *Management of Security Attributes*

Only authorized Hercules® AVR users have access to the functions of the TOE (F.IAUSER). These users are subject to the IMPORT_SFP information flow control security functional policy for the import of vulnerability scan data (F.IMPVADATA) and vulnerability remediation data (F.IMPREMDATA). Authorized users may also display the imported vulnerability data (F.DISPVADATA) and aggregate vulnerability information from multiple scans into a unified vulnerability picture for client systems (F.AGGVADATA). Authorized TOE users have the ability to manipulate all of the vulnerability and remediation data held by the TOE (F.MANAGEDATA).

FMT_MSA.3 *Static attribute initialization*

Only authorized Hercules® AVR users have access to the TOE for the purposes of initializing security attributes (F.IAUSER). The security attributes are used for mutual identification and authentication between the Hercules® AVR Server and the client machines (F.IACLIENT). The Hercules® AVR users are subject to the IMPORT_SFP information flow control security function policy for the import of vulnerability scan data (F.IMPVADATA) and vulnerability remediation data (F.IMPREMDATA). Authorized TOE users may specify alternative initial values to override default values when data is imported (F.MANAGEDATA).

FMT_MTD.1 *Management of TSF Data*

Only authorized Hercules® AVR users have access to the TOE (F.IAUSER). Only these users have the ability to manipulate (display, modify, delete, aggregate) vulnerability data (F.AGGVADATA, F.DISPVADATA, F.DISPSIG) remediation data (F.DISPPROF, F.MANAGEPROF, F.APPPROF) and client system vulnerability and remediation data (F.DISPVULN, F.DISPCLIENT, F.DISPCLIENTSTATUS, F.DISPREMSTATUS, F.SCHEDREM).

FMT_SMF.1 *Specification of Management Functions*

The TOE allows authorized users complete control of the vulnerability and remediation data for all client systems (F.MANAGEDATA). Users may create, edit and approve remediation profiles for client systems or groups of client systems (F.MANAGEPROF, F.APPPROF). Users may also schedule automatic remediation activity for client systems (F.SCHEDREM, F.PUSHREM). This allows users to remove specific vulnerabilities from specific client systems (F.REMCLIENT). If desired it is also possible is specific circumstances to roll back a previously applied remediation (F.ROLLBACK).

FMT_SMR.1 *Security Roles*

By default the TOE uses only one role; the Hercules® AVR user role. Members of this role have access to all of the functionality of the TOE. Additionally only individuals authorized as administrators by the underlying operating system are recognized as members of the Hercules® AVR user role (F.IAUSER). The TOE provides the capability to create custom roles to which individual users and groups of users may be assigned (F.MANAGEROLES). The ability to use specific features of the TOE such as the creation of user defined vulnerabilities may be assigned to custom roles.

FPT_ITT.1 *Basic Internal TSF Data Transfer Protection*

The TOE uses SSL to secure data transfers between the Administrator Console and the Remediation Server(s) (F.ENCRYPT, F.IACONSOLE, F.IAREMSVR). The TOE uses SSL (for Windows® clients) and OpenSSH (for Unix clients) to secure data transfers between a Remediation Server and client systems (F.ENCRYPT, F.IACLIENT, F.PUSHREM). These functions prevent the unauthorized disclosure and/or modification of TSF data.

FPT_RVM.1 *Non-bypassability of the TSP*

The TOE ensures that the TSP enforcement functions are invoked and successful before any function within the TSC is activated (F.RVM).

FPT_SEP.1 *TSF Domain Separation*

The TOE maintains a separate security domain for its own execution (F.DOMAINSEP). This protects the TOE from interference and tampering by untrusted subjects. The TOE also enforces separate security domains for each of the client systems being remediated (F.DOMAINSEP).

FPT_STM.1 *Reliable time stamps*

The audit functions of the TOE (F.AUDIT) use the reliable time stamp provided by the underlying operating system when recording audit records.

8.6 TOE ASSURANCE MEASURES RATIONALE

The Hercules® AVR product is designed to protect the TOE and its data from network attacks, to limit the system's use of network interfaces to those specified by the user, and to be simple enough for a knowledgeable system administrator to use. An assurance level of EAL 3, Methodically Tested and Checked, was selected as the threat to security is considered to be unsophisticated network attackers, and the data to be protected consists primarily of user-private data and system resources. An evaluation at this level provides a moderate level of independently assured security via a thorough investigation of the TOE and its development.

Table 8 provides a bi-directional mapping of Assurance Measures to Assurance Requirements, and is followed by a short discussion of how the Assurance Requirements are addressed by the corresponding Assurance Measures.

	ACM_CAP.3	ACM_SCP.1	ADO_DEL.1	ADO_IGS.1	ADV_FSP.1	ADV_HLD.2	ADV_RCR.1	AGC_ADM.1	AGD_USR.1	ALC_DVS.1	ATE_COV.2	ATE_DPT.1	ATE_FUN.1	ATE_IND.2	AVA_MSA.1	AVA_SOF.1	AVA_VLA.1
M.AUTH	X																
M.CONFIG	X	X															
M.DELIVER			X														
M.DESIGN					X	X	X										
M.DEVELOP										X							
M.DOCS								X	X						X		
M.ID	X																
M.SETUP				X													
M.TEST											X	X	X	X			

M.VULNER															X	X

Table 8 Mapping of Assurance Measures to Assurance Requirements

ACM_CAP.3 Authorisation Controls

Assurance Measure M.ID ensures that the TOE is uniquely identified and labelled with its identity. Assurance Measure M.CONFIG ensures that the TOE includes a configuration item list. Assurance Measure M.AUTH ensures that only authorised changes are permitted to the TOE. These measures combine to satisfy the requirements of ACM_CAP.3.

ACM_SCP.1 TOE CM Coverage

Assurance Measure M.CONFIG ensures that the TOE includes a configuration item list. The contents of this list ensure that the requirements of ACM_SCP.1 are met.

ADO_DEL.1 Delivery Procedures

Assurance Measure M.DELIVER ensures that the TOE includes documentation describing the delivery procedures for the TOE. This measure satisfies the requirements of ADO_DEL.1.

ADO_IGS.1 Installation, Generation and Startup Procedures

Assurance Measure M.SETUP ensures that the TOE includes documentation describing its secure installation, generation and startup. This measure satisfies the requirements of ADO_IGS.1.

ADV_FSP.1 Informal Functional Specification

Assurance Measure M.DESIGN ensures that the TOE design documentation includes an informal function specification. This measure satisfies the requirements of ADV_FSP.1.

ADV_HLD.2 Security Enforcing High Level Design

Assurance Measure M.DESIGN ensures that the TOE design documentation includes an informal high level design which includes; a description of the TSF in terms of subsystems, a description of the purpose and method of use of all interfaces to the subsystems and a description of the separation of the TOE into TSP enforcing and other subsystems. These features satisfy the requirements of ADV_HLD.2.

ADV_RCR.1 Informal Correspondence Demonstration

Assurance Measure M.DESIGN ensures that the TOE design documentation includes an informal correspondence demonstration between the TOE Summary Specification, the Functional Specification and the High Level Design. This measure satisfies the requirements of ADV_RCR.1.

AGD_ADM.1 Administrator Guidance

Assurance Measure M.DOCS ensures that the TOE documentation includes a user manual and online help system. Since all users of the TOE are also administrators (refer to assumption A.TOEUSER), this documentation acts as both User and Administrator guidance. This measure satisfies the requirements of AGD_ADM.1.

AGD_USR.1 User Guidance

Assurance Measure M.DOCS ensures that the TOE documentation includes a user manual and online help system. This measure satisfies the requirements of AGD_USR.1.

ALC_DVS.1 Identification of Security Measures

Assurance Measure M.DEVELOP ensures that the TOE documentation includes a description of the security measures for the TOE development environment. This measure satisfies the requirements of ALC_DVS.1.

ATE_COV.2 Analysis of Coverage

Assurance Measure M.TEST ensures that the TOE test documentation includes sufficient evidence to confirm that the developer has systematically tested the TOE against its functional specification and high level design. This measure satisfies the requirements of ATE_COV.2.

ATE_DPT.1 Testing: High Level Design

Assurance Measure M.TEST ensures that the TOE test documentation includes sufficient evidence to demonstrate that the TSF operates in accordance with its high level design. This measure satisfies the requirements of ATE_DPT.1.

ATE_FUN.1 Functional Testing

Assurance Measure M.TEST ensures that the TOE test documentation is sufficient to determine that the developer has functionally tested all TOE security functions. This measure satisfies the requirements of ATE_FUN.1.

ATE_IND.2 Independent Testing – Sample

Assurance Measure M.TEST ensures that the TOE test documentation is sufficient for the evaluator to repeat a sample of the developers functional testing in order to confirm the test results as well as develop independent tests of the TOE security functions. This measure satisfies the requirements of ATE_IND.2.

AVA_MSU.1 Examination of Guidance

Assurance Measure M.DOCS ensures that the TOE documentation includes guidance documentation. This documentation may be examined for misleading, unreasonable and conflicting guidance. This measure satisfies the requirements for AVA_MSU.1.

AVA_SOF.1 Strength of TOE Security Function Evaluation

Assurance Measure M.VULNER ensures that the TOE vulnerability analysis documentation includes a strength of TOE security function analysis for each mechanism identified in the ST as having a strength of TOE security function claim. This measure satisfies the requirements of AVA_SOF.1.

AVA_VLA.1 Developer Vulnerability Analysis

Assurance Measure M.VULNER ensures that the TOE vulnerability analysis documentation includes an analysis of obvious ways in which a user can violate the TOE security policies along with the disposition of these obvious vulnerabilities. This measure satisfies the requirements of AVA_VLA.1.