National Information Assurance Partnership

Common Criteria Evaluation and Validation Scheme Validation Report

SecureVue, Version 3.6.3 CP1

Report Number: **CCEVS-VR-VID10379-2013**

Dated: **May 20, 2013**

Version: **1.0**

National Institute of Standards and Technology
Information Technology Laboratory
100 Bureau Drive
Gaithersburg, MD 20899

National Security Agency
Information Assurance Directorate
9800 Savage Road STE 6940
Fort George G. Meade, MD 20755-6940

Table of Contents

List of Figures

1. Executive Summary

This Validation Report (VR) documents the evaluation and validation of the product SecureVue, Version 3.6.3 CP1.

This VR is not an endorsement of the IT product by any agency of the U.S. Government and no warranty of the IT product is either expressed or implied.

SecureVue from EiQ Networks is an IT security, risk and audit management platform that combines security information management (SIM) with governance, risk and compliance (GRC) to improve operational efficiency and reduce management complexity. Using an integrated model, SecureVue collects, correlates, archives, analyzes and reports on critical security and compliance data. Through end-to-end correlation, SecureVue transforms volumes of log, vulnerability, configuration, asset, performance, and flow data to automate incident identification and security breaches. Built-in network behavioral anomaly detection (NBA) automatically profiles flow data to identify anomalies. Additionally, a compliance library maps directly to specific regulations, best practices and control frameworks.

The TOE provides the following security functionality: auditing of security relevant events; TOE user identification and authentication; security role based access to management functions; trusted communication between components; cryptographic support; monitoring and management of network data for risk and compliance assessment; self-test of the TOE security functionality.

The TOE is intended for use in computing environments where there is a low level threat of malicious attacks. The assumed level of expertise of the attacker for all the threats is unsophisticated.

The evaluation was performed by the CygnaCom Common Criteria Testing Laboratory (CCTL), and was completed in April 2013. The information in this report is derived from the Evaluation Technical Report (ETR) and associated test reports, all written by the CygnaCom CCTL. The evaluation team determined that the product is Common Criteria version 3.1 R3 [CC] Part 2 extended and Part 3 conformant, and meets the assurance requirements of EAL 2 augmented with ALC_FLR.2 from the Common Methodology for Information Technology Security Evaluation, Version 3.1 R3, [CEM]. This Security Target does not claim conformance to a protection profile.

The evaluation and validation were consistent with National Information Assurance Partnership (NIAP) Common Criteria Evaluation and Validation Scheme (CCEVS) policies and practices as described on their web site www.niap-ccevs.org. The Security Target (ST) is contained within the document *SecureVue, Version 3.6.3 CP1 Security Target.*

2. Identification

Target of Evaluation: SecureVue, Version 3.6.3 CP1

Evaluated Software and Hardware:

SecureVue, Version 3.6.3 CP1 consisting of the following components:

- *Central Server*
  - *Web Based GUI*
- *Data Collector*
- *Host OS Agents (UNIX, Windows)*

Developer: EiQ Networks, Inc.

CCTL: CygnaCom Solutions

7925 Jones Branch Dr, Suite 5400

McLean, VA 22102-3321

Evaluators: Ms. Nancy Gow

Validation Scheme: National Information Assurance Partnership CCEVS

Validators: Mr. Bradford O'Neill, Dr. Patrick Mallett

CC Identification: Common Criteria for Information Technology Security Evaluation, Version 3.1 R3, July 2009

CEM Identification: Common Methodology for Information Technology Security Evaluation, Version 3.1 R3, July 2009

3. Security Policy

The TOE's security policy is expressed in the security functional requirements identified in Section 6.1 of the ST. Potential customers of this product should confirm that functionality implemented is suitable to meet the customers' requirements.

The following sections describe the TOE's security features.

3.1. Security Audit Functions

The Central Server generates individual audit records of security significant events and associates each auditable event with the identity of the TOE user account that caused the event. The TOE generated records are stored separately from the host OS's audit records. The TOE provides a decentralized auditing functionality. The Central Server stores the audit trail at the OS level within the SecureVue directory tree.

To view the audit records an administrator has several options in the Central Server's Web Based GUI to view all user activity. These viewing functions give the administrator the ability to custom query (search and sort) the audit data based on *User, Timestamp, and/or Activity.*

The administrators cannot modify or delete audit data through the TOE interfaces.

3.2. Identification and Authentication Functions

Each individual must be successfully identified and authenticated with a username and password by the TSF or by an authentication service in the Operational Environment that has been invoked by the TSF before access is allowed to the TOE. An administrator can add new user accounts to SecureVue by the following ways:

- Create a new user for native password handling (TOE authentication decision)
- Import Windows System Users (External authentication decision)
- Add Active Directory server (External authentication decision)
- Import Active Directory User (External authentication decision)
- Add RADIUS server (External authentication decision)

The Central Server is responsible for enforcing the I&A decision made natively or received from the configured external authentication mechanism. The OS, AD server, and RADIUS server are not in the scope but the TOE enforcement of the authentication decision and import services are.

For native password authentication the TOE collects the I&A information from the potential user over a secure channel (https://ServerIP entered into browser) via a pop-up (Java applet) window. The secure channel is established in the operating environment between the browser and the Apache/Microsoft IIS server (both of which are out of scope).

The TOE employs password masking during input, and a password policy that controls the password length and complexity when the user has been set to authenticate via Native Password handling. The password policy also includes hardcoded parameters for the implementation of a failed authentication handling (3 failed attempts = account lockout for 5 min.) mechanism and to force the re-authentication of a user after a period of non-activity to ensure login security.

The SecureVue Password Policy is hardcoded and NOT configurable. The hardcoded values determine the length and character sets that need to make up an acceptable password and is different for each role. The more privileged the role the longer the password length requirement.

The TSF maintains the following security attributes for each individual TOE user:

- **Username** (Login Name / Account Name)
- **Password** (Stored encrypted using AES DATABASE Key)
- **Authentication Mechanism** (Radius, AD, or native)
- **Enable/Disable account flag** (Disabled accounts cannot access the system)
- **Email ID** (Used for email notifications)
- **User Group** (Administrative Role / Security Role)
- **Device Group** (Used to restrict user to certain groups of devices)
- **Report Selection** (Used to restrict what reports a user can access based on role/group assignment)

3.3. Security Management Functions

The management functions for the Central Server are accessible through the Central Server's Web Based GUI.

The TOE maintains administrative roles that determine the access an account holder has to the management functions and TSF data. All users of the TOE have access to management functions and TSF data and are considered administrators. The administrative role is determined by the User Group attribute of an individual's account.

After the user has successfully authenticated, the TOE determines if the management function is available to that role. If the role does not have the privilege or permission the function is not activated (i.e. the Web Based GUI doesn't present the function).

The TOE supports 5 types of default user roles plus the ability to create custom roles:

- **Super Administrator:** There has to be one Super Administrator (also referred to as Super Admin) to manage the TOE. The Super Administrator is used to install the TOE. Once the TOE is installed, the functionality of the Super Administrator is the same as the Administrator. However, when this role is assigned to an "administrator" user, the TOE prevents that user account from being deleted.

- **Administrator:** An Administrator can manage entire application with exclusive rights to control, create, delete, and edit even other users with customized privileges. Users from this group have most rights in the Web Based GUI. Users from this group can also initiate FIPS SELF-TEST and re-generate Communication Key commands. Only one administrator can be assigned the Super Administrator role.
- **Power User:** Users in this group can be classified as read-only admins. They cannot manage Devices, Hosts, Groups, Users, Topology and Licenses. The Power User can create, edit, delete and view profiles, however, access to Collection-based policies and generation of file-based profiles is restricted.
- **User:** User accounts in this group can only generate all or few instant reports sections depending on the privileges assigned in the user policy. This role's access to reports and functions can only be customized by the Super Admin.
- **Alert User**: User accounts in this group have access to just the Alerts portal in the main console. Can only view, acknowledge, and clear alerts to which they have been granted access. Cannot edit, copy, delete, or create alerts, and cannot access the rule templates.
- **Custom User Roles:** Administrators can create custom users roles by assigning privileges and permissions to existing roles or completely new roles.

3.4. Protection of Security Functions

The Central Server performs a number of power-up and conditional self-tests to ensure proper operation of the cryptographic module. Power-up tests include cryptographic algorithm known answer tests and integrity tests. The integrity tests are performed using a HMAC-SHA-256 digest calculated over the object code of SecureVue. Power-up tests are run automatically when the cryptographic module is initialized. Additionally, power-up tests may be executed at any time by the administrator requesting the cryptographic module to force re-run of self-tests.

If the tests fail, a Log file is created giving brief description of FIPS Self-Test Suite results, and Transitions to a Power-OFF state.

3.5. Trusted Channel and Cryptographic Support Functions

The TSF includes a trusted communication infrastructure that provides trusted communication channels among its separately installed components. The 'trusted communication channel' ensures the two end points, (i.e., two components) are authenticated, their identity is associated to the data they transfer and that the data transferred is protected from modification and disclosure. The trusted communication channel between TOE components is established even if the components are installed on the same platform such as the Central Server and Data Collector can be installed on same platform.

Establishment of these trusted communications channels depend on the functionality of both the TOE (crypto module) and the Operational Environment (network infrastructure and host TCP/IP protocols)

SecureVue uses and provides the FIPS 140-2 validated (Certificate #1051) OpenSSL cryptographic module Version 1.2. The services used by SecureVue are Key Transfer, Communications, Database, File/Password-encryption, and Decryption.

Communications to the browser is support by the operational environment using Apache or Microsoft IIS Server. Apache includes the use of a separate instantiation of OpenSSL that is not part of the FIPS certified cryptographic module but is part of Apache installation. MS IIS uses the default MS crypto module provided with the OS. The trusted channel used between the browser and the Central Server (handshaking and cipher suite) uses FIPS certified algorithms.

3.6. Monitoring and Management of Network Functions

The TOE provides network monitoring and management of IT network assets for risk and compliance assessment. These functions include: scheduling the collection of network management and security data, storing uploaded collection data, evaluation of the collected data, and sending notifications to appropriate personnel for significant events in the assessment process.

The information security and event management, through real-time monitoring and concise reporting solely depends on the policies enforced for event data collection. SecureVue provides a visual interface to create and manage the policies for specific event data collection. An Administrator can create and enforce the event collection policies and policy templates for effective event management. There are also ready-to-use collection policies available in SecureVue.

The Central Server is responsible for the management of the collection policy. The Data Collector is responsible for the actual implementation of data collecting. An Agent (OSAgent) is an alternate way to collect host data for use in SecureVue. By installing the agent on an enterprise Windows/Linux asset, a user can collect Windows/Linux host data from that host. The Agent has the additional capability to monitor changes on folders, files, registry (Windows only) and USB devices in real-time.

***Note**: This host/asset could be considered hostile as the TOE administrator may not have direct control over this asset. This machine would be a multipurpose machine with non-administrative personnel having access and control over this machine.*

The analysis provided by the TOE is driven and governed by the same policies as the collection procedures. In SecureVue, a policy is a formal set of rules to define the course of action that the user needs to take under specific circumstances. A rule can dictate—which devices or hosts to consider, what event type to filter or negate, which entities with what values to add and so on. The user can associate a severity level to the Policy created. A policy is created on the customized device and/or host based rules or the existing rule templates. On implementation of a policy, the Administrator can choose to -- trigger an alert notification, or simply classify the Policy under an Event class by associating it to a report query. An Administrator can add, edit, copy or delete a Policy.

The analysis methodology include threshold verification, sequence matching, comparative (historical deltas), comparative against selected standards templates (for GRC Auditor function), and filtering based on policy or real time user input requests.

The administrator can configure a policy to send an alert upon indication of an unwanted pattern/activity happening in the network. When an alert is generated, it can be displayed on the Central Server's Web Based GUI and/or be sent as an email notification or SNMP trap.

3.7. Summary

3.7.1. Security functional Requirements

A list of the SFRs for the TOE follows.

***Note**: "_EXP" in the SFR ID indicates extended requirements. The ST must be consulted for the specifics of the _EXP requirements and the completions of the SFRs drawn from the CC.*

1	FAU_GEN.1	Audit data generation
2	FAU_GEN.2	User identity association
3	FAU_SAR.1	Audit review
4	FAU_SAR.3	Selectable audit review
5	FCS_CKM.1	Cryptographic key generation
6	FCS_CKM.4	Cryptographic key destruction
7	FCS_COP.1	Cryptographic operation
8	FIA_AFL.1	Authentication failure handling
9	FIA_ATD.1	User attribute definition
10	FIA_SOS.1	Verification of secrets
11	FIA_UAU_EXP.2	TSF user authentication before any action
12	FIA_UAU.5	Multiple authentication mechanisms
13	FIA_UAU.7	Protected authentication feedback
14	FIA_UID.2	User identification before any action
15	FMT_MTD.1	Management of TSF data
16	FMT_SMF.1	Specification of Management Functions
17	FMT_SMR.1	Security Roles
18	FPT_ITT_EXP.1	Explicit: Partial Inter-TSF trusted channel among distributed TOE components

19	FPT_TST_EXP.1	Explicit: TSF Self-testing
20	NMA_COL_EXP.1	Explicit: Asset data collection
21	NMA_EVL_EXP.1	Explicit: Asset data analysis and evaluation
22	NMA_NOT_EXP.1	Explicit: Security notifications

3.7.2. Operational Environment Objectives

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

1. The Operational Environment will provide an authentication service for user identification and authentication that can be invoked by the TSF to control a user's logical access to the TOE.

   **Note: OE.AuthService is only applicable to the TOE if is configured to use an external authentication service. (i.e. RADIUS Server)*

2. The administrator will ensure that there are no untrusted users and no untrusted software on the TOE component servers.
3. The TOE will be installed, configured and operated in a secure manner as outlined in the supplied guidance.
4. Personnel working as authorized administrators will be carefully selected and trained for proper operation of the system.
5. Those responsible for the TOE will ensure that those parts of the TOE critical to the security policy are protected from any physical attack.
6. The Operational Environment will provide a means for secure storage and protection of the TOE audit information from unauthorized users via the Operational Environment interfaces.
7. Users will ensure that their authentication data is held securely and not disclosed to unauthorized persons.
8. Those responsible for the TOE will ensure the communications between the TOE components and remote users are via a secure channel.
9. The Operational Environment will be configured by those responsible for the TOE to protect information stored in the database systems used by the TOE via the Operational Environment interfaces.
10. The Operational Environment will be configured by those responsible for the TOE to protect executable and data files used by the TOE via the Operational Environment interfaces.

11 The Operational Environment will, working in conjunction with the TOE, establish a trusted communications path which provides for protection of the data from modification or disclosure while being exchanged between TOE components.

12 The underlying operating system will provide reliable time stamps.

4. Assumptions and Clarification of Scope

4.1. Usage Assumptions

For secure usage, the operational environment must be managed in accordance with the documentation associated with the following EAL 2 assurance requirements.

a) AGD_OPE.1 Operational user guidance

b) AGD_PRE.1 Preparative procedures

c) ALC_DEL.1 Delivery procedures

4.2. Assumptions

The ST identifies the following assumptions about the use of the product:

1 It is assumed that one or more authorized administrators are assigned who are competent to manage the TOE and the security of the information it contains, trained for the secure operation of the TOE, and who can be trusted not to deliberately abuse their privileges so as to undermine security.

2 It is assumed that authorized TOE users are trusted to correctly install, configure and operate the TOE according to the instructions provided by the TOE documentation.

3 It is assumed that there will be no untrusted users and no untrusted software on the TOE component servers.

4 It is assumed that the TOE components critical to the security policy enforcement will be protected from unauthorized physical modification.

5 It is assumed that those responsible for the TOE will ensure the communications between the TOE components and remote users are protected to the level required for the operating environment.

6 It is assumed that those responsible for the TOE will ensure that data stored in the databases used by the TOE will be protected from unauthorized access via the IT Environment interfaces.

7 It is assumed that those responsible for the TOE will ensure executable and data files used by the TOE will be protected from unauthorized access via the IT Environment interfaces.

8 It is assumed that users will protect their authentication data.

4.3. Clarification of Scope

This section covers the limitations and clarifications of this evaluation. Note that:

1. This evaluated configuration satisfies the security claims made with the EAL2 level of assurance.
2. This evaluation only covers the specific version of the product identified in this document, and not any earlier or later versions released or in process.
3. As with EAL 2 evaluations, this evaluation did not specifically search for, nor seriously attempt to counter, vulnerabilities that were not "obvious" or vulnerabilities to objectives not claimed in the ST. The CEM defines an "obvious" vulnerability as one that is easily exploited with a minimum of understanding of the TOE, technical sophistication and resources.
4. The following are not included in the Evaluation Scope:

   TOE functionality considered out of scope
   - High availability option
   - Data Collector Configuration (DCConf.exe)
   - Distributed and tiered deployments

   OE software requirements/options provided on the installation disk:
   - Apache Server Version 2.2.22 with OpenSSL (different from crypto module in the TOE software used for trusted communication between TOE components)
   - Tomcat Server Version 7.0.26.0

5. The IT environment needs to provide the following capabilities:
   - Microsoft IIS Web Server (optionally used instead of Apache server) v7.0 minimum
   - Host OS for any of the TOE components
     - Network Protocols
   - Third party software loaded on TOE
     - Java (JRE) 1.6 or higher
     - MS-Office (to generate reports in WORD or EXCEL formats)
     - Adobe Acrobat Reader 6.0 or higher (to view reports in PDF format)
   - RADIUS Server
   - Active Directory Server
   - SNMP Server
   - SMTP Server
   - Any third party software in the IT Environment that supplies TOE with data
     - Profilers such as IDP, and NetFlow

    - Vulnerability scanners, such as Nessus
    - OS
    - Workflow Ticket management systems, such as Remedy
- Network infrastructure (switches, dns, dhcp, managed assets etc.)
- Host hardware for any of the TOE components

The ST provides additional information on the assumptions made and the threats countered.

5. Architectural Information

SecureVue provides effective real-time management of all log, vulnerability, configuration, asset, performance and Network behavior and Anomaly (NBA) data collected from network devices, systems and applications. Collected data is then normalized across disparate devices, aggregated into a database and correlated for monitoring, alerting, reporting and forensic tasks.

SecureVue can be installed in 3 deployment models: Standalone, Distributed, and Tiered. In the Standalone model the main components: the Central Server and Data Collector can be installed on the same physical hardware or on separate machines. The Distributed model introduces a third component called the Data Processor. The Data Processor is installed on a separate machine and is in-between the Central Server and the Data Processor. The Tiered model allows for multiple servers (Global, Regional, Local Servers and Data Processors to support large enterprise deployments. All modes support high availability configuration options that are not in scope of this evaluation.

Agents that are standalone executables to support the collection of information on a managed Windows, or UNIX node are installed directly on that managed node and are referred to as OS Agents.

The physical boundary of the TOE includes the entire product as commercially available from the developer.

The evaluated configuration of SecureVue is a standalone network deployment (no high availability) that includes the Central Server and Data Collector installed on separate hardware platforms, Host OS Agents (Window, and UNIX), and user documentation.

The platforms that house the Central Server and/or the Data Collector software are expected to be dedicated to the functionality of the TOE (i.e. non-TOE-supporting software should not be installed).

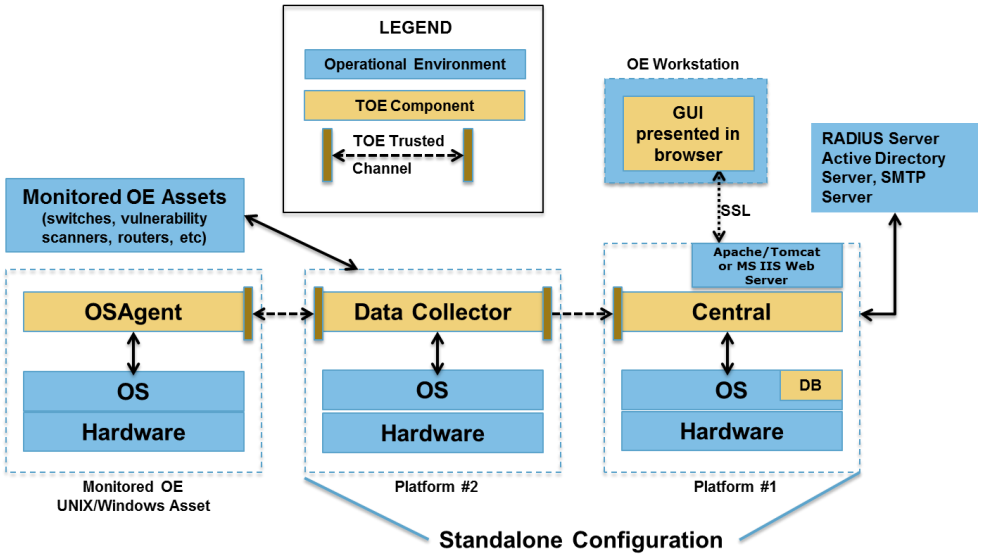

Figure 1: TOE Boundary

Central Server

The SecureVue Central Server is the nerve center of the solution performing all the data correlation and analytics, alert configuration, forensic analysis, GRC, and data archive management functions. The Central Server is responsible for the following security features: audit generation and review, management access control enforcement, identification and authentication (natively or by invoking an external mechanism), secure role based management via the Web Based GUI, protection of the TSF, trusted communication between components, trusted communications between the Central Server and a Browser for the Web Based GUI, management of the monitored network, risk and compliance assessment of the managed network.

This component is installed on its own platform as indicated in the figure. The platform and OS is responsible for protecting the stored audit and TOE executables.

Data Collector

The Data Collector interfaces between the Central Server and all the network devices, systems and applications within a SecureVue deployment. It is responsible for collecting log, vulnerability, configuration, asset, performance and NBA data automatically from all configured network devices, compressing them into delta files and sending to the Central Server for correlation, display, forensics, reporting and archiving. The Data Collector automatically updates the delta files (extracts of an original log file that only contains data that has been logged since the last update) to the Central Server on a regular basis without intervention from the administrator. The collected data is transferred to Central

Server in encrypted format by using Central Server's provided unique communication key.

The Data Collector is responsible for the following security features: collecting network information from specified assets, trusted communication with central server and agents. The Data Collector is operationally managed by the Central Server via the Central Server's Web Based GUI.

This component is installed on its own platform as indicated in the figure above

Agents

An Agent (OSAgent) is an alternate way to collect host data for use in SecureVue. By installing the agent on an enterprise Windows/Linux asset, a user can collect Windows/Linux host data from that host. The Agent has the additional capability to monitor changes on folders, files, registry (Windows only) and USB devices in real-time.

***Note**: This host/asset could be considered hostile as the TOE administrator may not have direct control over this asset. This machine would be a multipurpose machine with non-administrative personnel having access and control over this machine.*

The OSAgent polls the Data Collector every 5 minutes and in response, the Data Collector sends updates to OSAgent as requested (such as: adding/deleting/editing policies, changing run level, disabling agent etc). The collected agent data is transferred to Data Collector in encrypted format by using Data Collector provided unique communication key.

6. Documentation

The TOE is physically delivered to the End-User or downloaded from the vendor's website. The guidance is part of the TOE and is delivered on the installation media.

The following guidance documents are developed and maintained by EiQ Networks and delivered to the end user of the TOE:

- **EiQ Networks SecureVue 3.6.3 Deployment Guide**, 2013-Mar-06
- **EiQ Networks Release Notes: SecureVue®v3.6.0 Released on 05/29/2012,** 2012-May-29
- **EiQ Networks SecureVue 3.6 Upgrade Guide**, 2012-May-25
- **EiQ Networks SecureVue 3.6 User Guide**, 2012-May-14
- **EiQ Networks Release Notes: SecureVue®v3.6.3 Released on 12/28/2012,** 2012-Dec-28
- **SecureVue v3.6 CC Supplement Guide, 2013-Mar-28**

7. IT Product Testing

At EAL 2, the overall purpose of the testing activity is “independently testing a subset of the TSF, whether the TOE behaves as specified in the design documentation, and to gain confidence in the developer's test results by performing a sample of the developer's tests” (ATE_IND.2, 14.6.2.1 [CEM])

At EAL 2, the developer’s test evidence must “show the correspondence between the tests provided as evaluation evidence and the functional specification. However, the coverage analysis need not demonstrate that all TSFI have been tested, or that all externally-visible interfaces to the TOE have been tested. Such shortcomings are considered by the evaluator during the independent testing.” (ATE_COV.1, 14.3.1.3 [CEM])

This section describes the testing efforts of the vendor and the evaluation team.

The objective of the evaluator’s independent testing sub-activity is “to demonstrate that the security functions perform as specified. Evaluator testing includes selecting and repeating a sample of the developer tests” (ATE_IND.2, Independent testing – sample [CC]).

7.1. Developer Testing

The developer testing effort that is described in detail in the Developer Test Plan involved executing the test sets in the test configurations described in Section **Error! Reference source not found.Error! Reference source not found.**: **Error! Reference source not found.Error! Reference source not found.**.

7.1.1. Overall Test Approach and Results:

The developer's testing strategy was to define test cases that specified complete coverage of all security functions defined in the ST. These test cases were mapped to SFRs, TSFIs, and TOE Component listed in the ST, Functional Specification [FSP], and Common Criteria Test Coverage Document. After the test cases were defined, test procedures were written by the vendor’s development team to exercise each test case.

All of the developer test cases are manual, i.e. all test steps including setup and cleanup steps were performed by a user entering commands a terminal running the Web Based GUI. The tests were written to use the Web Based GUI to exercise the functions of the TOE.

The Wireshark packet analyzer was used for the FPT_ITT_EXP.1 tests to prove the encryption of data between the TOE components.

An EiQ Networks written executable, SocketTest Client.exe, was used to create a client socket and to send plain text for the FPT_ITT_EXP.1 tests to prove that transmission of plain text between TOE components will be rejected.

7.1.2. Depth and Coverage

All developer test cases test TOE security functions by stimulating an external interface.

Although the developer tests are performed using the Web Based GUI, the evaluator determined that the test cases as described in the test documentation adequately exercise the internal interfaces.

The developer provided a test plan, test procedures and test evidence consisting of screen shots of the actual results from the execution of the tests:

- The developer's test plan covered all of the security relevant behavior of each Security Function in the ST.
- The developer wrote test procedures for 100% of the cases identified in the Common Criteria Test Coverage Document.
- The Developer executed all of their test procedures and provided the actual results.
- The developer's test procedures covered 100% of the TOE SFRs claimed in the Security Target.
- The developer's test procedures covered all but 2 of the External TSF Interfaces.
- The developer's test procedures covered 100% of the Internal Subsystem Interfaces.

7.1.3. Results

The evaluator checked the developer’s test procedures and the test evidence and found that the expected test results are consistent with the actual test results provided. For each test case examined, the evaluator checked the expected results in the test procedures with the actual results provided in the test evidence and found that the actual results were consistent with the expected results.

Given the Evaluation Assurance level (EAL 2), the evaluator determined that Vendor’s TOE testing is adequate. The vendors TOE testing exercises all security functions identified in the Functional Specification.

7.2. Evaluator Independent Testing

The evaluator performed the following activities during independent testing:

- Installation of the TOE in its evaluated configuration (AGD_PRE.1)
- Execution the Developer’s Functional Tests (ATE_IND.2)
- Evaluator-Defined Functional Testing (ATE_IND.2)
- Vulnerability/Penetration Testing (AVA_VAN.2)

Two platforms were provided by the vendor for testing. One had the TOE already installed along with the groups, policies, agents and other prerequisites needed to run all

of the vendor's functional tests, the other was a clean platform that was used for the evaluator's installation of the TOE. The vendor explained that setting up all the prerequisites by the evaluator would take at least an additional four to five days. Therefore, the evaluator made the decision to use this preinstalled platform for testing after confirming that it was indeed in the evaluated configuration as specified in the ST.

The evaluator confirmed that the operational environment for both platforms conformed to the configuration specified in the ST and the vendor's test plan.

When the evaluator arrived on-site at EiQ Networks, however, the vendor explained that the version of the TOE that was pre-installed and the one that would be installed by the evaluator was SecureVue v3.6.3 CP1 rather than version 3.6.2.6 which was the version used for TVOR. Patches were made to the product since TVOR to fix bugs and vulnerabilities according to the documented flaw remediation procedures. The evaluator examined the release notes and found that there were no changes to the security functionality or the structure of the product that would require changes to the SFRs in the ST or the ADV evidence documentation.

Installation was successful and the TOE was installed in the evaluated configuration as specified in the ST. At the end of the installation the evaluator identified the TOE components reference numbers (i.e. version numbers) using the procedures outlined in the CM documentation and found that they matched the new evaluated version of the TOE: v3.6.3 CP1.

7.2.1. Execution of the Developer's Functional Tests

The evaluator initially chose 20% (72 out of 364 tests) of the Developer Functional tests to be run to provide:

- A representative sample of all Developer Functional tests
- At least one test per SFR
- At least one test per TSFI (except for those that are subject for independent testing)
- Complete coverage of all Subsystems and Internal Interfaces

Particular care was taken to choose tests that exercise the administrative functionality, access control and authentication & identification functionality of the TOE.

In actuality, 217 out of 364 (60%) of the developer's functional tests were rerun during the on-site testing. (Other functionality of the TOE, such as sending email alerts and running reports were exercised, however, formal test steps were not followed and these tests are not included in this count.)

During testing, the parameter values used in commands were changed on an ad-hoc basis from the values documented in the developer's functional test steps to ensure the full functionality of each interface.

The tests were run by the evaluator or by the vendor at the evaluator's direction. The evaluator took notes and screenshots during the entire testing process.

The sampling of the developer's functional test cases were executed after the TOE was installed in the evaluated configuration consistent with the Security Target

Only one test did not perform as expected. Vendor Test 96256 which is a test of FAU_SAR.1 was written to verify that audit data could be viewed by both admin and non-admin users. However, running the test showed that only users with the administrator role had access to the audit data. The FAU_SAR.1 SFR was updated in the ST as a result of this testing.

Test 96155 (Zeroization) could not be initially run by the evaluator. The problem was reported to the development team, who sent an email stating the documented procedure was missing one step:

- Open command prompt Run As Administrator (Right click on cmd available in Start menu to use Run As Administrator option)

Once the corrected steps were followed, the evaluator was able to successfully run the test.

7.2.2. Evaluator-Defined Functional Testing

The evaluator-defined tests were devised to augment the developer's functional tests in order to exercise functionality in greater depth than the developer tests provided. Because of the extensive coverage of the vendor tests, the following five evaluator-defined tests were defined and run to cover functionality not exercised in the vendor tests.

1	**Create User with no Permissions**	The purpose of this test case is to verify that creation of a user fails if a no permissions are assigned to them
2	**Import AD User with Bad Credentials**	The purpose of this test case is to verify that the TOE will not import an AD user with bad credentials
3	**Run Topology Test using ICMP**	The purpose of this test case is to verify that the Topology functionality of the TOE will work with ICMP (in addition to SNMP)
4	**CC Collection Policy Test**	Verify the data collection policies with the audit log and monitoring capabilities of the TOE (suggested by validators as a TVOR Action Item)
5	**Custom User Test**	The purpose of this test case is to exercise the custom user functionality of the TOE (suggested by TVOR Action Item)

In addition to these tests, throughout the running of the vendor tests the evaluator used input parameters (names, policy parameters …) other than those specified in the vendor's test procedure documentation on an ad-hoc basis.

The test environment and configuration were the same as for the developer's functional testing. No special tools were used for the evaluator-defined functional testing.

All evaluator-defined tests passed with no comments. All vendor tests run with input other than that documented by the vendor ran as expected.

7.2.3. Vulnerability/Penetration Testing

The Vulnerability / Penetration tests covered hypothesized vulnerabilities and potential misuse of guidance.

All evidence deliverables were considered for identifying potential vulnerabilities. An analysis of the design documentation identified no specific vulnerabilities. The FSP and TDS documents describe the TOE at a high level that is consistent with the EAL 2 assurance requirements.

The evaluator searched for publicly known vulnerabilities that affected the eIQ product line and the SecureVue product. The evaluator searched the CVE database for eIQ products, the TOE and the Operational Environment components that are included with the TOE. The search for publicly known vulnerabilities also included a search for vulnerabilities that affected similar products that could potentially be applicable to the TOE. No applicable public vulnerabilities were found.

Based on the evaluator's vulnerability analysis, the evaluator did not find vulnerabilities that are applicable to the TOE in its operational environment. However, the evaluator identified penetration test scenarios that can be applied to TOE in its operational environment.

The following were performed as ad-hoc tests during the on-site testing. No formal test procedures were written for these tests.

1	**Buffer overflow attacks**	Enter large quantities of input through the Web Based GUI to see if the TOE enters an insecure state.
2	**Cross-site scripting (XSS) attacks**	Enter active URL and/or HTML date via the Web Based GUI to see if the TOE enters an insecure state.
3	**Bad input data in the Web GUI may create conditions that the application cannot handle**	Enter bad data such as alphabetic data in numeric fields, bad dates … to see if it creates an insecure state.

The following tests were developed by the evaluator on-site after gaining better knowledge of the TOE.

1	**Power User Audit Access**	The purpose of this test case is to verify that only admin users have access to the user activity logs This test was developed after Vendor Test 96256 was run and the evaluator discovered that only admin users should have access to the audit logs. Other user interface options than that described in the vendor test were tested to see if this limitation could be by-passed.
2	**Browser Refresh after Logout**	The purpose of this test case is to verify that after logout, a user will have to be re-authenticated to gain access to the TOE
3	**Port 8080 Access**	The purpose of this test case is to verify that an unauthorized application cannot access the TOE through the Central Server's port 8080.
4	**SSH Handshake**	Wireshark should be used to examine the SSH handshake and verify that FIPS approved algorithms were selected.
5	**N-Stalker Runs**	Run N-Stalker in accordance with the instructions provided in "Steps to generate a N-Stalker Report with SecureVue" document. Try various scanning options and generate reports

The actual results of the penetration tests were recorded as captured screen shots and N-Stalker reports.

For the ad-hoc testing of the user interface described above, the evaluator entered invalid data, large copied text files and active URLs in the text entry boxes of the user interface. In all cases an error message such as "text limit is 30 characters" or "invalid value" were displayed and the security of the TOE was not compromised.

Penetration Tests 1 and 2 failed during on-site testing. The failures were reported to the Vendor POC who informed the development team in India.

These bugs were corrected according to the vendor's flaw remediation procedures. New tests were developed to test the corrections and entered into the Test Link system used for testing CM. Proof of the remediation was sent to the evaluator. These fixes were installed as a patch to the TOE software and are included in the evaluated version of the TOE.

Various scan options were used to run N-Stalker to search for vulnerabilities in the TOE. N-Stalker was run according to the instructions in the N-Stalker user guidance. Only one high level vulnerability was found:

> **Webserver is vulnerable to SSL MITM renegotiation attack.**
>
> Your webserver has a vulnerable SSL software that might allow a malicious user to perform man-in-the-middle (MiTM) attacks against your application's users.

This vulnerability (and other medium/low vulnerabilities that are related to it) was already discovered by the Vendor. Fixes for this are included in the evaluated version of the TOE.

8. Evaluated Configuration

The TOE was tested the following test bed components:

SecureVue v3.6.3 CP1 Central Server specification:

Processor: Intel(R) Xeon(R) CPU X3430 @ 2.40GHz 2.39 GHz
Memory: 8 GB Storage: 250 GB on 7200 RPM SATA drives
Operating System: 64 bit Windows Server 2008 R2
Java: Java (JRE) 1.6 r30
Microsoft Office 2007 is required to generate Microsoft Word or Excel reports.

SecureVue v3.6.3 CP1 Data Collector specification:

Processor: AMD Sempron @ 2.20 GHz
Memory: 2 GB RAM
Storage: 80+160 HDD
Operating System: 64bit Windows Server 2008 R2

SecureVue v3.6.3 CP1 Windows Agent specification:

Processor: Intel Dual Core @ 2.7 GHz
Memory: 4 GB
Storage: 160 GB on 7200 RPM SATA drives
Operating System: 32bit Windows Server 2003.

SecureVue v3.6.3 CP1 UNIX/Linux Agent specification:

Processor: Intel Xeon @ 2.3 GHz
Memory: 2 GB
Storage: 100 GB on 7200 RPM SATA drives
Operating System: CentOS 5 (x86_64)

The Operational Environment included the following test bed components

Category	**Device Type and Version**
Firewall	Cisco ASA v8.x
Router	Cisco IOS v12.x
IDS / IPS	SourceFire v1.x
Vulnerability Scanner	Nessus v4.x
Web Servers	Microsoft IIS v7.5
Gateway	BlueCoat Proxy SG v5.2
Server and Desktop OS	Windows Servers 2008 Windows7 Redhat Linux ES 5.x
OS Agents	CentOS 5.x

	Windows 2003
Applications	Microsoft SQL Enterprise Edition 2005
Others	Microsoft Active Directory 6.1.7

Both Apache and Microsoft IIS were used in testing.

9. Results of Evaluation

A verdict for an assurance component is determined by the resulting verdicts assigned to the corresponding evaluator action elements. The evaluation was conducted based upon version 3.1 R3 of the CC and the CEM.

The Evaluation Team assigned a Pass, Fail, or Inconclusive verdict to each work unit of each EAL 2 augmented with ALC_FLR.2 assurance component. For Fail or Inconclusive work unit verdicts, the Evaluation Team advised the developer of issues requiring resolution or clarification within the evaluation evidence. In this way, the Evaluation Team assigned an overall Pass verdict to the assurance component only when all of the work units for that component had been assigned a Pass verdict.

The details of the evaluation are recorded in the Evaluation Technical Report (ETR), which is controlled by CygnaCom CCTL.

Below lists the assurance requirements the TOE was required meet to be evaluated and pass at Evaluation Assurance Level 2 augmented with ALC_FLR.2. The following components are taken from CC part 3. The components in the following section have no dependencies unless otherwise noted.

- ADV_ARC.1 Security architecture description
- ADV_FSP.2 Security-enforcing functional specification
- ADV_TDS.1 Basic design
- AGD_OPE.1 Operational user guidance
- AGD_PRE.1 Preparative procedures
- ALC_CMC.2 Use of a CM system
- ALC_CMS.2 Parts of the TOE CM coverage
- ALC_DEL.1 Delivery procedures
- ALC_FLR.2 Flaw reporting procedures
- ASE_CCL.1 Conformance claims
- ASE_ECD.1 Extended components definition
- ASE_INT.1 ST Introduction
- ASE_OBJ.2 Security objectives
- ASE_REQ.2 Derived security requirements
- ASE_SPD.1 Security problem definition
- ASE_TSS.1 TOE summary specification
- ATE_COV.1 Evidence of coverage
- ATE_FUN.1 Functional testing

- ATE_IND.2 Independent testing – sample
- AVA_VAN.2 Vulnerability analysis

The evaluators concluded that the overall evaluation result for the target of evaluation is Pass. The evaluation team reached PASS verdicts for all applicable evaluator action elements and consequently all applicable assurance components.

- The TOE is CC Part 2 Extended
- The TOE is CC Part 3 Conformant.
- The validators reviewed the findings of the evaluation team, and have concurred that the evidence and documentation of the work performed support the assigned rating.

10. Validators Comments/Recommendations

None.

11. Security Target

SecureVue, Version 3.6.3 CP1 Security Target is compliant with the Specification of Security Targets requirements found within Annex B of Part 1of the CC.

12. Glossary

12.1. Product Specific Acronyms and Terminology

The following are product specific acronyms and terms. Not all are used in this document.

API	Application Programming Interface that uses Remote Registry to interface
CPMI	Check Point Management Interface
DAS	Direct-attached-storage system
Device Group	A collection of devices with a unique name that can then be assigned to a user or user group
Devices	Any network asset such as a host, router, switch, firewall etc.
Forensics	Forensics analysis involves recording and analysis of network events in order to discover the source of security attacks or other problem incidents.
FTP	File Transfer Protocol
FTPS	(also known as *FTP Secure* and *FTP-SSL*) is an extension to the commonly used File Transfer Protocol (FTP) that adds support for the Transport Layer Security (TLS) and the Secure Sockets Layer (SSL) cryptographic protocols.
ICMP	The Internet Control Message Protocol is one of the core protocols of the Internet Protocol Suite. It is chiefly used by networked computers' operating systems to send error messages—indicating, for instance, that a requested service is not available or that a host or router could not be reached. Notable exception to this is the Ping and TraceRoute user commands
IP	Internet Protocol is a protocol used for communicating data across packet-switched network.
IT Governance	IT Governance Establishes Decision Structures And Tracking Mechanisms
IT Risk Management	IT Risk Management Helps Mitigate Adverse Effects And Identifies Opportunities
IT Compliance	IT Compliance Establishes And Monitors IT Controls (Auditor function: compare real vs set of rules that determine compliance)
MIB	Management information base is a type of database used to manage the devices in a communications network. It comprises a collection of objects in a (virtual) database used to manage entities (such as routers and switches) in a network.
NBA	Network behavior and Anomaly

NAS	Network-Attached-Storage
Network Management	Network Management covers a wide variety of definitions. For this document, it is scoped to these terms. Security: Ensuring that the network is protected from unauthorized users. Performance: Eliminating bottlenecks in the network. Reliability: Making sure the network is available to users and responding to hardware and software malfunctions. Also see IT Governance, IT Risk, and IT Compliance
Policies	A Policy is a systematic set of statements to govern the upcoming decisions and actions of the user.
Profiles	A profile is a set of instructions identifying the locations of the device logs, how data must be accessed, the method followed to analyze data, how IP addresses must be resolved, and customization of reports.
SAN	Storage-Area-Network
SIM	Security Information Management
SSH	Secure Shell is a network protocol that allows data to be exchanged using a secure channel between two networked devices.
SMTP	Simple Mail Transfer Protocol is an Internet standard for electronic mail transmission across Internet Protocol networks.
SNMP	Simple Network Management Protocol a communication protocol between management stations, such as consoles, and managed objects (MIB objects), such as routers, gateways, and switches, makes use of MIBs.
SSL	Secure Sockets Layer, now Transport Layer Security, a communications protocol
TCP	Transmission Control Protocol is one of the core protocols of the Internet Protocol Suite. TCP is one of the two original components, with Internet Protocol (IP), of the suite, so that the entire suite is commonly referred to as *TCP/IP*.
TLS	Transport Layer Security and its predecessor, Secure Sockets Layer, are cryptographic protocols that provide security and data integrity for communications over networks such as the Internet. TLS and SSL encrypt the segments of network connections at the Transport Layer end-to-end.
Telnet	A network protocol used on the Internet or local area networks to provide a bidirectional interactive communications facility.

Topology	Topology is the schematic description of the arrangement of a network, including its nodes and connecting lines.
User Group	Equivalent of user roles. A user is assigned to a user group (administrator, power-user, user) which then dictates to the TSF which functions and TSF data is available for the authenticated user to access. One user, assigned to the administrator user group, is also selected for the role of Super Admin.
WMI	Windows Management Instrumentation (WMI) is the infrastructure for management data and operations on Windows-based operating systems.

12.2. CC Specific Acronyms and Terminology

This section defines the CC-specific acronyms and terms. Not all of these are used in this document.

Assurance	Grounds for confidence that an entity meets its security objectives.
Attack potential	The perceived potential for success of an attack, should an attack be launched, expressed in terms of a threat agent's expertise, resources and motivation.
Augmentation	The addition of one or more assurance component(s) to a package.
Authentication data	Information used to verify the claimed identity of a user.
Authorised user	A user who may, in accordance with the SFR, perform an operation.
CC	Common Criteria [for IT Security Evaluation]
CEM	Common Methodology for Information Technology Security Evaluation
Class	A grouping of families that share a common focus.
Component	The smallest selectable set of elements on which requirements may be based.
Connectivity	The property of the TOE that allows interaction with IT entities external to the TOE. This includes exchange of data by wire or by wireless means, over any distance in any environment or configuration.
Dependency	A relationship between components such that if a requirement based on the depending component is included in a PP, ST or package, a requirement based on the component that is depended upon must normally also be included in the PP, ST or package..
EAL	Evaluation Assurance Level
Element	An indivisible security requirement.

Evaluation	Assessment of a PP, an ST, or a TOE against defined criteria.
Evaluation Assurance Level (EAL)	A package consisting of assurance components from Part 3 that represents a point on the CC predefined assurance scale.
Evaluation authority	A body that implements the CC for a specific community by means of an evaluation scheme and thereby sets the standards and monitors the quality of evaluations conducted community.
Evaluation scheme	The administrative and regulatory framework under which the CC is applied by an evaluation authority within a specific community.
Extension	The addition to an ST or PP of functional requirements not contained in Part 2 and/or assurance requirements not contained in Part 3 of the CC.
External entity	Any entity (human or IT) outside the TOE that interacts (or may interact) with the TOE.
Family	A grouping of components that share security objectives but may differ in emphasis or rigor.
Formal	Expressed in a restricted syntax language with defined semantics based on well-established mathematical concepts.
Identity	A representation (e.g. a string) uniquely identifying an authorized user, which can either be the full or abbreviated name of that user or a pseudonym.
Internal communication channel	A communication channel between separated parts of TOE.
Internal TOE transfer	Communicating data between separated parts of the TOE.
Inter-TSF transfers	Communicating data between the TOE and the security functions of other trusted IT products.
IT	Information Technology
Iteration	The use of the same component to express two or more distinct requirements.
Object	A passive entity in the TOE, that contains or receives information, and upon which subjects perform operations.
Organizational security policies	A set of security rules, procedures, or guidelines imposed (or presumed to be imposed) now and/or in the future by an actual or hypothetical organization in the operational environment.

OSP	Organizational Security Policy
Package	A named set of either functional or assurance requirements (e.g. EAL 3).
PP	Protection Profile
Protection Profile (PP)	An implementation-independent statement of security needs for a TOE type.
Prove	This term refers to a formal analysis in its mathematical sense. It is completely rigorous in all ways. Typically, "prove" is used when there is a desire to show correspondence between two TSF representations at a high level of rigor.
Refinement	The addition of details to a component.
Role	A predefined set of rules establishing the allowed interactions between a user and the TOE.
SAR	Security Assurance Requirement
Secure state	A state in which the TSF data are consistent and the TSF continues correct enforcement of the SFRs.
Security attribute	A property of subjects, users (including external IT products), objects, information, sessions and/or resources that is used in defining the SFRs and whose values are used in enforcing the SFRs.
Security Function Policy (SFP)	A set of rules describing specific security behaviour enforced by the TSF and expressible as a set of SFRs.
Security objective	A statement of intent to counter identified threats and/or satisfy identified organization security policies and/or assumptions.
Security Target (ST)	An implementation-dependent statement of security needs for a specific identified TOE.
Selection	The specification of one or more items from a list in a component.
Semiformal	Expressed in a restricted syntax language with defined semantics.
SFP	Security Function Policy
SFR	Security Functional Requirement
ST	Security Target
Subject	An active entity in the TOE that performs operations on objects.
Target of Evaluation (TOE)	A set of software, firmware and/or hardware possibly accompanied by guidance.
TOE	Target of Evaluation

TOE resource	Anything useable or consumable in the TOE.
TOE Security Functions (TSF)	A set consisting of all hardware, software, and firmware of the TOE that must be relied upon for the correct enforcement of the TSP.
Transfers outside TSF	TSF mediated communication of data to entities not under control of the TSF.
Transfers outside TSF	TSF mediated communication of data to entities not under control of the TSF.
Trusted channel	A means by which a TSF and a remote trusted IT product can communicate with necessary confidence.
Trusted path	A means by which a user and a TSF can communicate with necessary confidence.
TSC	TSF Scope of Control
TSF	TOE Security Functions
TSF data	Data created by and for the TOE that might affect the operation of the TOE.
TSF interface (TSFI)	A means by which external entities (or subjects in the TOE but outside of the TSF) supply data to the TSF, receive data from the TSF and invoke services from the TSF.
TSFI	TOE Security Functions Interface
TSP	TOE Security Policy
User	See external entity
User data	Data created by and for the user that does not affect the operation of the TSF

13. Bibliography

URLs

[1] Common Criteria Evaluation and Validation Scheme (CCEVS): (http://www.niap-ccevs.org/cc-scheme).

[2] CygnaCom Solutions CCTL (http://www.cygnacom.com).

CCEVS Documents

[1] Common Criteria for Information Technology Security Evaluation - Part 1: Introduction and general model, July 2009 Version 3.1 Revision 3 Final, CCMB-2009-07-001.

[2] Common Criteria for Information Technology Security Evaluation - Part 2: Security functional components, July 2009 Version 3.1 Revision 3 Final, CCMB-2009-07-002.

[3] Common Criteria for Information Technology Security Evaluation - Part 3: Security assurance components, July 2009, Version 3.1 Revision 3 Final, CCMB-2009-07-003.

[4] Common Methodology for Information Technology Security Evaluation - Evaluation methodology, July 2009, Version 3.1 Revision 3 Final, CCMB-2009-07-004