National Information Assurance Partnership

Common Criteria Evaluation and Validation Scheme Validation Report

IBM

Red Hat Enterprise Linux

Version 4, Update 1

Report Number: CCEVS-VR-06-0009

Dated: 15 January 2006

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 6740
Fort George G. Meade, MD 20755-6740

ACKNOWLEDGEMENTS

Validation Team

The Aerospace Corporation
Columbia, MD

Evaluation Team

atsec Information Security Corporation
Austin, TX

Table of Contents

1. EXECUTIVE SUMMARY

This report documents the NIAP validators' assessment of the evaluation of IBM Red Hat Enterprise Linux (RHEL) Version 4, Update 1. It presents the evaluation results, their justifications, and the conformance results. This validation report 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.

The evaluation was performed by the atsec Information Security Corporation, and was completed during January 2006. The information in this report is largely derived from the Evaluation Technical Report (ETR) and associated test report, both written by the CCTL. The evaluation determined the product to be **Part 2 extended, Part 3 conformant,** and to meet the requirements of **EAL4 augmented by ALC_FLR.3.** Additionally, the TOE was shown to satisfy the requirements of the Controlled Access Protection Profile (CAPP), Issue 1.d, 8 October 1999.

Red Hat Enterprise Linux (RHEL) is a general-purpose, multi-user, multi-tasking operating system. As such, it provides a platform for a wide variety of arbitrary applications. Red Hat AS is available on a broad range of systems, from departmental servers to multi-processor enterprise servers; RHEL WS is available on workstations and small servers.

The evaluation covers a potentially distributed, but closed network of IBM xSeries, pSeries, iSeries, and eServer 325 servers running the evaluated configurations of the TOE.

The validation team monitored the activities of the evaluation team, provided guidance on technical issues and evaluation processes, reviewed successive versions of the Security Target, reviewed selected evaluation evidence, reviewed test plans, reviewed intermediate evaluation results (i.e., the CEM work units), and reviewed successive versions of the evaluation technical report (ETR) and test report. The validation team determined that the evaluation team showed that the product satisfies all of the functional requirements and assurance requirements defined in the Security Target (ST) for a CAPP-compliant, EAL4 evaluation. Therefore, the validation team concludes that the CCTL findings are accurate, and the conclusions justified.

2. IDENTIFICATION

The CCEVS is a joint National Security Agency (NSA) and National Institute of Standards and Technology (NIST) effort to establish commercial facilities to perform trusted product evaluations. Under this program, security evaluations are conducted by commercial testing laboratories called Common Criteria Testing Laboratories (CCTLs) using the Common Evaluation Methodology (CEM) for Evaluation Assurance Level (EAL) 1 through EAL 4 in accordance with National Voluntary Laboratory Assessment Program (NVLAP) accreditation.

The NIAP Validation Body assigns Validators to monitor the CCTLs to ensure quality and consistency across evaluations. Developers of information technology products desiring a security

evaluation contract with a CCTL and pay a fee for their product's evaluation. Upon successful completion of the evaluation, the product is added to NIAP's Validated Products List.

Table 1 provides information needed to completely identify the product, including:

- The Target of Evaluation (TOE): the fully qualified identifier of the product as evaluated;
- The Security Target (ST), describing the security features, claims, and assurances of the product;
- The conformance result of the evaluation;
- The Protection Profile to which the product is conformant;
- The organizations and individuals participating in the evaluation.

Table 1: Evaluation Identifiers

Item	Identifier
Evaluation Scheme	United States NIAP Common Criteria Evaluation and Validation Scheme
Target of Evaluation	IBM Red Hat Enterprise Linux, Version 4 Update 1
Protection Profile	Controlled Access Protection Profile (CAPP), Issue 1.d, 8 October 1999.
Security Target	*Red Hat Enterprise Linux Version 4 Update 1 Security Target for CAPP Compliance;* Version 2.6, 2 November 2005
Evaluation Technical Report	*Evaluation Technical Report a Target of Evaluation: Red Hat Enterprise Linux Version 4 Update I AS, and Red Hat Enterprise Linux Version 4 Update 1 WS.* Version 3.0, 16 December 2005
Conformance Result	CC V 2.2, Part 2 extended, Part 3 conformant, EAL 4 augmented by ALC_FLR.3, and CAPP-compliant
Sponsor	IBM
Developer	IBM and Red Hat
Evaluators	atsec GmbH
Validators	The Aerospace Corporation

3. SECURITY POLICY

3.1. Access Control

Red Hat Enterprise Linux implements Discretionary Access Control (DAC) through the use of standard UNIX permission bits and the POSIX standard Access Control Lists (ACLs). A Discretionary Access Control policy requires mechanisms whereby the access of users (i.e., subjects) to system resources and data (i.e., objects) can be controlled on the basis of user identity, role, and explicit permissions. Mechanisms that implement a DAC policy provide the capability for users to specify the how their personal data objects are to be shared.

Permission bits are associated with objects and specify the permissions (typically, READ, WRITE, EXECUTE) for a specific user, the user's group, and all others (i.e., "world"). Access Control Lists provide the same functionality relative to granting specific permissions, but are considerably more flexible in that they can identify a number of group affiliations for a single user.

The standard UNIX DAC mechanism is permission bits, as is the case with RHEL. However, RHEL implements ACLs as an extended permission mechanism, available at the discretion of the file owner; ACLs are supported only for file system objects.[1]

3.2. I&A

Each user must have a unique identity (i.e., username plus password), and be authenticated prior to obtaining resources and services from the TOE. Note, however, that in a networked environment, user identities are unique to a server, and are neither known globally nor are universally unique. That is, each server maintains its own set of users and their associated passwords and attributes. A user that has access to more than one server on a network will have a different user identity, and possibly different attributes, on each server for which access is authorized.

Users can change their own passwords. However, an administrator can define the following constraints for the authentication process:

- Maximum duration of a password (i.e., time-to-live);
- Minimum time allowed between password changes;
- Minimum password length;
- Number of days warnings are displayed prior to password expiration;
- Allowed number of consecutive unsuccessful login attempts;
- Disallowed passwords (i.e., the TOE retains a history of recently-used passwords to prevent users from cycling previously-used passwords).

The proper parameters for each of these choices is defined for the evaluated configuration

3.3. Auditing

The TOE audit mechanism allows the generation of audit records for security-related events, and allows the administrator to configure the audit mechanism to define which events are to be captured and which users are to be audited; it is also possible for the administrator to identify specific users that are not to be audited.

Each audit record contains event-specific information, and identifies whether the request that caused the event was successful or failed. An audit record consists of a standard header that includes the following information:

[1] See Section 6.2.4 of the ST for a fuller discussion of the DAC mechanisms and the algorithm by which access determinations are made.

- A unique audit identifier;
- The LoginID of the user who caused the audit record to be generated;
- The Effective User ID of the user at the time the record was generated;
- Date and time the audit record was generated;
- Type of event.

Audit records are stored in ASCII format, and can be searched through the use of the standard UNIX/LINUX *grep* tool.

3.4. Object Reuse

Although the TOE supports several different types of objects, each is managed by the system such that no pre-existing content is provided to users to whom objects are allocated. That is, whenever an object (e.g., buffers, memory extents, disk space) is allocated to a user process, it is managed such that any data that had previously been in the object (i.e., from an earlier process) is unavailable to the new process.

In short, memory pages are initialized to all zeroes when allocated to a process, IPC objects are also initialized to all zeroes, file system objects are created with no content (with the exception of directories and symbolic links).[2]

4. ASSUMPTIONS

4.1. Usage Assumptions

Although there are additional assumptions stated in the Security Target[3], the primary conditions are that:

- The TOE is located within controlled facilities and is protected from unauthorized physical access;
- TOE hardware and software are protected from unauthorized modification;
- All authorized users possess authorization for at least some of the data managed on the TOE;
- The TOE operates in a relatively benign environment;
- Unencrypted communications paths, and communications paths within the controlled facility are protected from unauthorized physical access.

[2] A more complete discussion of object reuse for each of the various object types is contained in Section 6.2.4 of the ST.

[3] See section 3.1 of the ST

4.2. Clarification of Scope

The TOE includes the hardware platform (see Section 8) and all the code that enforces the policies identified (see Section 3). TOE also includes secure communications functions; i.e., SSH V2 and SSL V3).

The administrator tools are not considered to be part of the TSF. The administrator uses the commands that are provided by RHEL for system management; these are utilities that execute in untrusted user space, and are protected by the normal O/S mechanisms that prevent user processes from interfering with each other. Note that system management tools do not enforce TOE security policies, and that the TSF checks that the caller is authorized to invoke the requisite system calls and has the access rights to the objects being accessed.

5. ARCHITECTURAL INFORMATION

The TOE is a multi-user, multi-tasking operating system which can support multiple users simultaneously. A fundamental protection mechanism is the memory management and virtual memory support provided by the hardware. This provides a domain (i.e., supervisor state) in which only the kernel executes.

The TSF comprises two major components: kernel software and trusted processes.

The kernel software executes in supervisor state, which is supported by the memory management mechanism in the hardware. The memory management mechanism insures that only kernel code can execute in the supervisor state—from which all of memory may be accessed—and also serves to protect the kernel code from external tampering. The kernel implements file and I/O services, which provides access to files and devices. The kernel also implements:

- Named pipes
- Unnamed pipes
- Signals
- Semaphores
- Shared memory
- Message queues
- Internet domain sockets
- Unix domain sockets.

The trusted processes, which provide the remainder of the TSF, are referred to as “non-kernel TSF” services because they run in user state; they execute in the same hardware domain as user applications. These are protected from external tampering through the process management and memory virtualization mechanisms that implement per-process address spaces and prevent processes from interfering with each other. They are also protected from unauthorized access by the access control mechanisms of the TSF. The primary non-kernel TSF services are:

- Identification and authentication

- Network application layer services
- Configuration and management commands requiring root privileges.

6. DOCUMENTATION

The TOE is delivered with the following user documentation:

- CAPP EAL4 Evaluated Configuration Guide for Red Hat Enterprise Linux on IBM Hardware, Version 1.14;
- Installation Guide for the IBM eServer iSeries and IBM eServer pSeries Architectures, Version RHEL4;
- Installation Guide for the IBM S/390 and IBM eServer zSeries Architectures, Version RHEL4;
- Installation Guide for the x86, Itanium, and AMD64 Architectures, Version RHEL4;
- Red Hat Enterprise Linux 4 Reference Guide, Version RHEL4;
- Red Hat Enterprise Linux 4 System Administration Guide, Version RHEL4;
- Red Hat Enterprise Linux 4 Security Guide, Version RHEL4.

7. IT PRODUCT TESTING

7.1. Sponsor Testing

Testing is performed by both RedHat and IBM. However, only the tests performed by the sponsor (i.e., IBM) were considered applicable to this evaluation.

The majority of the tests cases are executed in the Linux Testing Project (LTP) environment. The sponsor also developed test cases for exercising the audit subsystem. These test cases were integrated into the LTP environment and included in the test suite that was run by the sponsor. Additionally, the sponsor developed tests for the *at* command, and for exercising the ACL functionality. Manual tests were also developed and included in the test suite. Where necessary, the sponsor adopted existing test cases (i.e., changing some of the internal structure of the test cases) to more accurately reflect and exercise the current TOE.

In the case of OpenSSL tests were developed based on the test suite from the OpenSSL developers.

The sponsor provided mappings of each test case to the relevant TSF interface (TSFI), interface specification (i.e., FSP), and high-level design description (i.e., HLD). The evaluators identified a number of security-relevant internal interfaces (i.e., between subsystems and not reflected to the external, user interface) that were defined in the HLD, and ascertained that these were also exercised by the test suite.

The evaluators ascertained that the testing was complete and fairly comprehensive, covering both explicit functionality as well as error conditions (e.g., invalid parameters, invalid credentials).

7.2. Evaluator Testing

As an integral component of testing, the evaluator installed and configured the TOE on each of the platforms, and verified that the configuration for each test TOE was consistent with the ST.

Because the majority of the sponsor's tests can be run automatically the evaluators executed the entire automated test suite on all platforms.

The sponsor's test suite was judged to be quite complete and comprehensive, and thus the evaluator needed to design relatively few additional tests. However, some additional test cases were developed and executed for:

- examining some of the TOE security functions in more detail than the sponsor-supplied test cases (e.g., Object Reuse, protection of audit records, password quality checks);
- examining aspects not covered by developer testing (e.g., verification of ACL support in the archival tool, system reaction to absent security-relevant configuration options);
- augmenting the testing of selected functions where the sponsor-supplied testing was deemed to be insufficiently broad.

7.3. Vulnerability Analysis and Testing

The evaluator reviewed the developer's vulnerability analysis. The developer's analysis was fairly extensive, including a review of a large number of published flaws for Linux. Some of the potential attacks were eliminated as being not valid within the presumed constraints of the evaluated configuration (e.g., network vulnerabilities would not apply because of the connectivity constraints articulated in the ST, which assume a benign, controlled network). Some of the published vulnerabilities were eliminated because they represented attacks against utilities that are not part of the evaluated TOE,[4] and others had been fixed in subsequent releases, and thus were no longer applicable. The developer identified the residual vulnerabilities; those that had not been fixed and that were considered applicable to the TOE configuration.

Evaluator analysis resulted in no additional attack paths being identified. To test claims that previously-discovered attacks had been fixed, some testing was done by the evaluators, resulting in the conclusion that the developer's fixes were effective in eliminating the flaws.

Of the potential attacks that were determined to be residual vulnerabilities, for the most part these are attacks that require skills that exceed the assumptions of the evaluation; they would require a "high attack potential." Others are attacks of the nature of viruses and "trojan horses" that take place in the user domain, and thus require care on the part of users regarding executables that they introduce into their own work spaces.

[4] This implies an obligation on the part of the TOE administrators to install the TOE as directed by the Evaluated Configuration Guide, and to pay careful attention to privileges granted to any additional software that is installed on the TOE (covered in section 4.4 of the configuration guide.

In short, the evaluators determined that the vulnerability analysis performed by the developer to be satisfy the requirements for such analysis, and the conclusions justified. Additionally, evaluator testing verified that fixes that had been implemented to correct previously-detected flaws were effective.

8. EVALUATED CONFIGURATION[5]

The evaluated and tested configurations are:

- IBM xSeries systems, based on Intel Xeon EM64T processor (Work Station and Server);
- IMB eServer BladeCenter systems based on the Intel Xeon EM64T processor (Work Station and Server);
- IBM xSeries x800, z900, z890, z990, executing in a z/VM5.1 virtual machine (Server only);
- IBM iSeries systems based on the POWER5 processor with pSeries LPAR and the OS/400 service partition (Server only);
- IBM pSeries based on the POWER5 processor with pSeries LPAR (Server only);
- IBM eServer systems based on the AMD Opteron processor (Server only).

9. RESULTS OF THE EVALUATION[6]

The evaluation team determined the product to be **CC Part 2 extended, CC Part 3 conformant, CAPP conformant,** and to meet the requirements of **EAL 4 augmented by ALC_FLR.3**. In short, the product satisfies the security technical requirements specified in *IBM RedHat Enterprise Linux Version 4 Update 1 Security Target for CAPP Compliance,* Version 2.6, 2005-11-02.

10. VALIDATOR COMMENTS

There are no validator comments.

11. SECURITY TARGET

The ST, *IBM RedHat Enterprise Linux Version 4 Update 1 Security Target for CAPP Compliance,* Version 2.6, 2005-11-02 is included here by reference.

[5] For more complete information on the evaluated configurations, see Section 2.4 of the Security Target.

[6] The terminology in this section is defined in CC Interpretation 008, specifying new language for CC Part 1, section/Clause 5.4.

12. LIST OF ACRYONYMS

CC	Common Criteria
CCEVS	Common Criteria Evaluation and Validation Scheme
CCTL	Common Evaluation Testing Laboratory
CEM	Common Evaluation Methodology
EAL	Evaluation Assurance Level
ETR	Evaluation Technical Report
NIAP	National Information Assurance Partnership
NIST	National Institute of Standards & Technology
NSA	National Security Agency
PP	Protection Profile
ST	Security Target
TOE	Target of Evaluation
TSF	TOE Security Function
TSFI	TOE Security Function Interface

13. BIBLIOGRAPHY

[1] Common Criteria for Information Technology Security Evaluation – Part 1: Introduction and general model; Version2.2, January 2004.

[2] Common Criteria for Information Technology Security Evaluation – Part 2: Security functional requirements; Version 2.2, January 2004 dated August 1999, Version 2.1.

[3] Common Criteria for Information Technology Security Evaluation – Part 3: Security assurance requirements; Version 2.2, January 2004.

[4] Common Evaluation Methodology for Information Technology Security Evaluation; Version 2.2, January 2004.

[5] Red Hat Enterprise Linux Version 4 Update 1 Security Target for CAPP Compliance, Version 2.6, 2005-11-02;

[6] Evaluation Technical Report for a Target of Evaluation, Red Hat Enterprise Linux Version 4 Update 1 AS, and Red Hat Enterprise Linux Version 4 Update 1 WS, Version 3.0, 2005-12-16;

[7] ATE: Developer Testing Work Package 13 RHEL Update 1, Version 2.0, 2005-11-06;

[8] IND: Installation and Independent Testing, Work Package 14, RHEL4 Update 1, Version 1.0, 2005-09-28.

[9] VLA: Vulnerability Assessment Work Package 15, RHEL4 U1; Version 1.0, 2005-10-25