Federal Office
for Information Security

Certification Report

BSI-DSZ-CC-0724-2012

For

Red Hat Enterprise Linux, Version 5.6
Virtualization with KVM

from

Red Hat, Inc.

BSI - Bundesamt für Sicherheit in der Informationstechnik, Postfach 20 03 63, D-53133 Bonn
Phone +49 (0)228 99 9582-0, Fax +49 (0)228 9582-5477, Infoline +49 (0)228 99 9582-111

Bundesamt
für Sicherheit in der
Informationstechnik

Deutsches IT-Sicherheitszertifikat

erteilt vom Bundesamt für Sicherheit in der Informationstechnik

BSI-DSZ-CC-0724-2012

Operating System

Red Hat Enterprise Linux, Version 5.6 Virtualization with KVM

from	Red Hat, Inc.
Functionality:	Product specific Security Target Common Criteria Part 2 extended
Assurance:	Common Criteria Part 3 conformant EAL 4 augmented by ALC_FLR.3

Common Criteria
Recognition
Arrangement

Common Criteria

The IT product identified in this certificate has been evaluated at an approved evaluation facility using the Common Methodology for IT Security Evaluation (CEM), Version 3.1 for conformance to the Common Criteria for IT Security Evaluation (CC), Version 3.1.

This certificate applies only to the specific version and release of the product in its evaluated configuration and in conjunction with the complete Certification Report.

The evaluation has been conducted in accordance with the provisions of the certification scheme of the German Federal Office for Information Security (BSI) and the conclusions of the evaluation facility in the evaluation technical report are consistent with the evidence adduced.

This certificate is not an endorsement of the IT product by the Federal Office for Information Security or any other organisation that recognises or gives effect to this certificate, and no warranty of the IT product by the Federal Office for Information Security or any other organisation that recognises or gives effect to this certificate, is either expressed or implied.

Bonn, 20 April 2012

For the Federal Office for Information Security

SOGIS IT SECURITY CERTIFIED

Bernd Kowalski L.S.
Head of Department

Bundesamt für Sicherheit in der Informationstechnik
Godesberger Allee 185-189 - D-53175 Bonn - Postfach 20 03 63 - D-53133 Bonn
Phone +49 (0)228 99 9582-0 - Fax +49 (0)228 9582-5477 - Infoline +49 (0)228 99 9582-111

This page is intentionally left blank.

Preliminary Remarks

Under the BSIG[1] Act, the Federal Office for Information Security (BSI) has the task of issuing certificates for information technology products.

Certification of a product is carried out on the instigation of the vendor or a distributor, hereinafter called the sponsor.

A part of the procedure is the technical examination (evaluation) of the product according to the security criteria published by the BSI or generally recognised security criteria.

The evaluation is normally carried out by an evaluation facility recognised by the BSI or by BSI itself.

The result of the certification procedure is the present Certification Report. This report contains among others the certificate (summarised assessment) and the detailed Certification Results.

The Certification Results contain the technical description of the security functionality of the certified product, the details of the evaluation (strength and weaknesses) and instructions for the user.

[1] Act on the Federal Office for Information Security (BSI-Gesetz - BSIG) of 14 August 2009, Bundesgesetzblatt I p. 2821

Contents

This page is intentionally left blank.

A Certification

1 Specifications of the Certification Procedure

The certification body conducts the procedure according to the criteria laid down in the following:

- BSIG[2]
- BSI Certification Ordinance[3]
- BSI Schedule of Costs[4]
- Special decrees issued by the Bundesministerium des Innern (Federal Ministry of the Interior)
- DIN EN 45011 standard
- BSI certification: Procedural Description (BSI 7125) [3]
- Common Criteria for IT Security Evaluation (CC), Version 3.1[5] [1]
- Common Methodology for IT Security Evaluation, Version 3.1 [2]
- BSI certification: Application Notes and Interpretation of the Scheme (AIS) [4]

2 Recognition Agreements

In order to avoid multiple certification of the same product in different countries a mutual recognition of IT security certificates - as far as such certificates are based on ITSEC or CC - under certain conditions was agreed.

2.1 European Recognition of ITSEC/CC – Certificates (SOGIS-MRA)

The SOGIS-Mutual Recognition Agreement (SOGIS-MRA) Version 3 became effective in April 2010. It defines the recognition of certificates for IT-Products at a basic recognition level and in addition at higher recognition levels for IT-Products related to certain technical domains only.

The basic recognition level includes Common Criteria (CC) Evaluation Assurance Levels EAL1 to EAL4 and ITSEC Evaluation Assurance Levels E1 to E3 (basic). For higher recognition levels the technical domain Smart card and similar Devices has been defined. It includes assurance levels beyond EAL4 resp.E3 (basic). In Addition, certificates issued for Protection Profiles based on Common Criteria are part of the recognition agreement.

---

2 Act on the Federal Office for Information Security (BSI-Gesetz - BSIG) of 14 August 2009, Bundesgesetzblatt I p. 2821

3 Ordinance on the Procedure for Issuance of a Certificate by the Federal Office for Information Security (BSI-Zertifizierungsverordnung, BSIZertV) of 07 July 1992, Bundesgesetzblatt I p. 1230

4 Schedule of Cost for Official Procedures of the Bundesamt für Sicherheit in der Informationstechnik (BSI-Kostenverordnung, BSI-KostV) of 03 March 2005, Bundesgesetzblatt I p. 519

5 Proclamation of the Bundesministerium des Innern of 12 February 2007 in the Bundesanzeiger dated 23 February 2007, p. 3730

As of September 2011 the new agreement has been signed by the national bodies of Austria, Finland, France, Germany, Italy, The Netherlands, Norway, Spain, Sweden and the United Kingdom. Details on recognition and the history of the agreement can be found at https://www.bsi.bund.de/zertifizierung.

The SOGIS-MRA logo printed on the certificate indicates that it is recognised under the terms of this agreement by the nations listed above.

2.2 International Recognition of CC – Certificates (CCRA)

An arrangement (Common Criteria Recognition Arrangement) on the mutual recognition of certificates based on the CC Evaluation Assurance Levels up to and including EAL 4 has been signed in May 2000 (CCRA). It includes also the recognition of Protection Profiles based on the CC.

As of September 2011 the arrangement has been signed by the national bodies of: Australia, Austria, Canada, Czech Republic, Denmark, Finland, France, Germany, Greece, Hungary, India, Israel, Italy, Japan, Republic of Korea, Malaysia, The Netherlands, New Zealand, Norway, Pakistan, Republic of Singapore, Spain, Sweden, Turkey, United Kingdom, United States of America. The current list of signatory nations and approved certification schemes can be seen on the website: http://www.commoncriteriaportal.org.

The Common Criteria Recognition Arrangement logo printed on the certificate indicates that this certification is recognised under the terms of this agreement by the nations listed above.

3 Performance of Evaluation and Certification

The certification body monitors each individual evaluation to ensure a uniform procedure, a uniform interpretation of the criteria and uniform ratings.

The product Red Hat Enterprise Linux, Version 5.6 Virtualization with KVM has undergone the certification procedure at BSI.

The evaluation of the product Red Hat Enterprise Linux, Version 5.6 Virtualization with KVM was conducted by atsec information security GmbH. The evaluation was completed on 30 March 2012. The atsec information security GmbH is an evaluation facility (ITSEF)[6] recognised by the certification body of BSI.

For this certification procedure the sponsor and applicant is: Red Hat, Inc..

The product was developed by: Red Hat, Inc..

The certification is concluded with the comparability check and the production of this Certification Report. This work was completed by the BSI.

4 Validity of the Certification Result

This Certification Report only applies to the version of the product as indicated. The confirmed assurance package is only valid on the condition that

- all stipulations regarding generation, configuration and operation, as given in the following report, are observed,

[6] Information Technology Security Evaluation Facility

- the product is operated in the environment described, where specified in the following report and in the Security Target.

For the meaning of the assurance levels please refer to the excerpts from the criteria at the end of the Certification Report.

The Certificate issued confirms the assurance of the product claimed in the Security Target at the date of certification. As attack methods evolve over time, the resistance of the certified version of the product against new attack methods needs to be re-assessed. Therefore, the sponsor should apply for the certified product being monitored within the assurance continuity program of the BSI Certification Scheme (e.g. by a re-certification). Specifically, if results of the certification are used in subsequent evaluation and certification procedures, in a system integration process or if a user's risk management needs regularly updated results, it is recommended to perform a re-assessment on a regular e.g. annual basis.

5 Publication

The product Red Hat Enterprise Linux, Version 5.6 Virtualization with KVM has been included in the BSI list of the certified products, which is published regularly (see also Internet: https://www.bsi.bund.de and [5]). Further information can be obtained from BSI-Infoline +49 228 9582-111.

Further copies of this Certification Report can be requested from the developer[7] of the product. The Certification Report may also be obtained in electronic form at the internet address stated above.

In case of changes to the certified version of the product, the validity can be extended to the new versions and releases, provided the sponsor applies for assurance continuity (i.e. re-certification or maintenance) of the modified product, in accordance with the procedural requirements, and the evaluation does not reveal any security deficiencies.

[7] Red Hat, Inc.
Varsity Drive
NC 27506 Raleigh
USA

This page is intentionally left blank.

B Certification Results

The following results represent a summary of

- the Security Target of the sponsor for the Target of Evaluation,
- the relevant evaluation results from the evaluation facility, and
- complementary notes and stipulations of the certification body.

1 Executive Summary

The TOE is Red Hat Enterprise Linux Version 5.6 with additional packages as listed in table 2.

Red Hat Enterprise Linux (RHEL) is a highly-configurable Linux-based operating system which has been developed to provide a good level of security as required in commercial environments. This evaluation focuses on the use of RHEL to provide a KVM virtualization environment. RHEL provides the host system for the virtual machine environment and manages the virtual machines using the KVM technology. In addition, RHEL provides management interfaces to administer the virtual machine environment as well as full auditing of user and administrator operations.

The KVM technology separates the runtime environment of virtual machines from each other. The Linux kernel operates as the hypervisor to the virtual machines but provides a normal computing environment to administrators of the virtual machines. Therefore, the Linux kernel supports the concurrent execution of virtual machines and regular applications. RHEL uses the processor virtualization support to ensure that the virtual machines execute close to the native speed of the hardware.

In addition to the separation of the runtime environment, RHEL also provides system-inherent separation mechanisms to the resources of virtual machines. This separation ensures that the large software component used for virtualizing and simulating devices executing for each virtual machine cannot interfere with each other. Using the SELinux multi-category mechanism, the virtualization and simulation software instances are isolated. The virtual machine management framework configures SELinux multi-category settings transparently to the administrator. The TOE Security Functional Requirements are implemented by the following TOE Security Functions:

- Auditing
- Cryptographically secured communication channels
- Packet filter
- Identification and Authentication
- Discretionary Access Control
- Authoritative Access Control
- Virtual machine environments
- Security Management

The Security Target [6] is the basis for this certification. It is not based on a certified Protection Profile.

The TOE Security Assurance Requirements (SAR) are based entirely on the assurance components defined in Part 3 of the Common Criteria (see part C or [1], Part 3 for details). The TOE meets the assurance requirements of the Evaluation Assurance Level EAL 4 augmented by ALC_FLR.3.

The TOE Security Functional Requirements (SFR) relevant for the TOE are outlined in the Security Target [6], chapter 6.2. They are selected from Common Criteria Part 2 and some of them are newly defined. Thus the TOE is CC Part 2 extended.

The TOE Security Functional Requirements are implemented by the following TOE Security Functionalities:

TOE Security Functionality	Addressed issue
Auditing	The Lightweight Audit Framework (LAF) is designed to be an audit system for Linux compliant with the requirements from Common Criteria. LAF is able to intercept all system calls as well as retrieving audit log entries from privileged user space applications. The subsystem allows configuring the events to be actually audited from the set of all events that are possible to be audited.
Cryptographically secured communication channels	The TOE provides cryptographically secured communication to allow remote entities to log into the TOE. For interactive usage, the SSHv2 protocol is provided. In addition, the access to the virtual machine administration tool as well as the virtual machine consoles can be protected using a SSHv2-based tunnel.
Packet filter	The TOE provides a stateless and stateful packet filter for regular IP-based communication. Layer 2 (IP) and layer 3 (TCP, UDP, ICMP) network protocols can be controlled using this packet filter. To allow virtual machines to communicate with the environment, the TOE provides a bridging functionality. Ethernet frames routed through bridges are controlled by a separate packet filter which implements a stateless packet filter for the TCP/IP protocol family as well as VLAN filtering.
Identification and Authentication	User identification and authentication in the TOE includes all forms of interactive login (e.g. using the SSH protocol or log in at the local console) as well as identity changes through the su command. These all rely on explicit authentication information provided interactively by a user.
Discretionary Access Control	DAC allows owners of named objects to control the access permissions to these objects. These owners can permit or deny access for other users based on the configured permission settings. The DAC mechanism is also used to ensure that untrusted users cannot tamper with the TOE mechanisms.
Authoritative Access Control	The TOE supports authoritative access control based on the following concepts: • SELinux categories are attached to virtual machines and its resources. The access control policy enforced using these categories grant virtual machines access to resources if the category of the virtual machine is identical to the category of the accessed resource. • Users cannot interfere with these labels. The TOE uses SELinux with an appropriate SELinux policy to enforce the authoritative access control.
Virtual machine environments	The TOE implements the host system for virtual machines. It acts as a hypervisor which provides an environment to allow other operating systems execute concurrently.
Security Management	The security management facilities provided by the TOE are usable by authorized users and/or authorized administrators to modify the configuration of TSF.

Table 1: TOE Security Functionalities

For more details please refer to the Security Target [6], chapter 6.2.

The assets to be protected by the TOE are defined in the Security Target [6], chapter 3.1.1. Based on these assets the TOE Security Problem is defined in terms of Assumptions, Threats and Organisational Security Policies. This is outlined in the Security Target [6], chapter 3.

This certification covers the following configurations of the TOE: The evaluated configuration is documented in the Evaluated Configuration Guide ([8]). It is based on Red Hat Enterprise Linux 5.6 (RHEL 5.6) with additional packages as listed in table 2.

The vulnerability assessment results as stated within this certificate do not include a rating for those cryptographic algorithms and their implementation suitable for encryption and decryption (see BSIG Section 9, Para. 4, Clause 2).

The certification results only apply to the version of the product indicated in the certificate and on the condition that all the stipulations are kept as detailed in this Certification Report. This certificate is not an endorsement of the IT product by the Federal Office for Information Security (BSI) or any other organisation that recognises or gives effect to this certificate, and no warranty of the IT product by BSI or any other organisation that recognises or gives effect to this certificate, is either expressed or implied.

2 Identification of the TOE

The Target of Evaluation (TOE) is called:

Red Hat Enterprise Linux, Version 5.6 Virtualization with KVM

The following table outlines the TOE deliverables:

No	Type	Identifier	Release	Form of Delivery
1.	SW	Red Hat Enterprise Linux 5.6 (RHEL 5.6)	Release 5, update 6	Download
2.	SW	Evaluation package RPM EAL4_RHEL5.6(cc-eal4-config-rhel56-0.13-1.noarch.rpm)	0.13.1	Download
3.	DOC	EAL4 Evaluated Configuration Guide for Red Hat Enterprise Linux with KVM support on IBM hardware	1.4	Download
4.	SW	Updates required: openssh-4.3p2-72.el5_6.3 for x86_64 architecture openssh-clients-4.3p2-72.el5_6.3 for x86_64 architecture openssh-server-4.3p2-72.el5_6.3 for x86_64 architecture libvirt-0.8.2-15.el5_6.3 for IA-32 and x86_64 architecture libvirt-python-0.8.2-15.el5_6.3 for IA-32 and x86_64architecture selinux-policy-2.4.6-300.el5_6.1 noarch package selinux-policy-devel-2.4.6-300.el5_6.1 noarch package selinux-policy-mls-2.4.6-300.el5_6.1 noarch package selinux-policy-targeted-2.4.6-300.el5_6.1 noarch package	see package names	Download

Table 2: Deliverables of the TOE

2.1 Overview of Delivery Procedure

The TOE is delivered from the developer, Red Hat, using the Red Hat delivery mechanism. There are several download components as shown in 'Table 2: Deliverables of the TOE' above.

RHEL 5.6 is delivered via the Red Hat Network (RHN), an online retrieval system provided by the developer. The packages are built by the Red Hat Release Engineering Group and immediately signed using the Red Hat PGP private Key (the public key is widely distributed and available). ISO images are created and SHA-256 checksums of the images are generated. The ISO images for the release are transferred to a staging area on the web server hosting the RHN using SSH. The SHA-256 checksums for the images are verified to ensure that the image has not been modified. The image is then moved to the public download area and the SHA-256 checksum is checked again to verify that the

image has not been modified. Customers download the ISO images and are advised to verify the checksums and the signatures.

The additional evaluation package contains the kickstart installation program and configuration files, as well as the Evaluated Configuration Guide [8]. The package is securely provided by the developer, reviewed and built into an RPM by the Team Lead for the Security Technologies Team,signed by Release Engineering using the signing key referenced above, and electronically delivered by Red Hat's FTP site. Customers who download the package are advised to verify the signature.

2.2 Identification of the TOE by the User

The customer can identify the TOE packages in the download sites by appropriate labeling (i.e.,the packages/ISO images for RHEL 5.6 are labeled RHEL5U6, and the evaluation package is labeled EAL4_RHEL5.6. Following installation, the user can verify by looking at the content of/etc/release that the installed version is "Red Hat Enterprise Linux Server release 5.6".

3 Security Policy

The Security Policy is expressed by the set of Security Functional Requirements and implemented by the TOE. It covers the following issues:

- The users of the TOE shall be held accountable for their security-relevant actions within the TOE.
- Authority shall only be given to users who are trusted to perform the actions correctly.
- When using SSH with key-based authentication, organizational procedures must exist that ensure users protect their private SSH key component against its use by any other user.

4 Assumptions and Clarification of Scope

The Assumptions defined in the Security Target [6] and some aspects of Threats and organisational Security Policies are not covered by the TOE itself. These aspects lead to specific security objectives to be fulfilled by the TOE-Environment. The following topics are of relevance:

- Those responsible for the TOE are competent and trustworthy individuals, capable of managing the TOE and the security of the information it contains.
- If the TOE relies on remote trusted IT systems to support the enforcement of its policy, those systems provide the functions required by the TOE and are sufficiently protected from any attack that may cause those functions to provide false results.
- Those responsible for the TOE must establish and implement procedures to ensure that information is protected in an appropriate manner. In particular:
    - All network and peripheral cabling must be approved for the transmittal of the most sensitive data held by the system. Such physical links are assumed to be adequately protected against threats to the confidentiality and integrity of the data transmitted.

  - DAC protections on security-relevant files (such as audit trails and authentication databases) shall always be set up correctly.
  - Users are authorized to access parts of the data managed by the TOE and are trained to exercise control over their own data.
- Those responsible for the TOE must establish and implement procedures to ensure that the hardware, software and firmware components that comprise the system are distributed,installed and configured in a secure manner supporting the security mechanisms provided by the TOE.
- Authorized users of the TOE must ensure that the comprehensive diagnostics facilities provided by the product are invoked at every scheduled preventative maintenance period.
- Those responsible for the TOE must ensure that those parts of the TOE critical to enforcement of the security policy are protected from physical attack that might compromise IT security objectives. The protection must be commensurate with the value of the IT assets protected by the TOE.
- Those responsible for the TOE must ensure that procedures and/or mechanisms are provided to assure that after system failure or other discontinuity, recovery without a protection (security) compromise is achieved.
- The remote trusted IT systems implement the protocols and mechanisms required by the TSF to support the enforcement of the security policy.

  These remote trusted IT systems are under the same management domain as the TOE, are managed based on the same rules and policies applicable to the TOE, and are physically and logically protected equivalent to the TOE.

5 Architectural Information

The TOE is structured in much the same way as many other operating systems, especially Unix-type operating systems. It consists of a kernel, which runs in the privileged state of the processor and provides services to applications (which can be used by calling kernel services via the system call interface). Direct access to the hardware is restricted to the kernel, so whenever an application wants to access hardware like disk drives, network interfaces or other peripheral devices, it has to call kernel services. The kernel then checks if the application has the required access rights and privileges and either performs the service or rejects the request.

The kernel is also responsible for separating the different user processes. This is done by the management of the virtual and real memory of the TOE which ensures that processes executing with different attributes cannot directly access memory areas of other processes but have to do so using the inter-process communication mechanism provided by the kernel as part of its system call interface.

The TSF of the TOE also include a set of trusted processes, which when initiated by a user with a system, call operate with extended privileges. The programs that represent those trusted processes on the file system are protected by the file system discretionary access control security function enforced by the kernel.

In addition the execution of the TOE is controlled by a set of configuration files, which are also called the TSF database. Also those configuration files are protected by the file system discretionary access control security function enforced by the kernel.

The kernel acts as a hypervisor for the virtual machine support of the TOE. It uses the virtualization support of the underlying processor to provide virtual machines with the required kernel support in KVM and user space support via libvirt.

Normal users – after they have been successfully authenticated by a defined trusted process – can start untrusted applications where the kernel enforces the security policy of the TOE when those applications request services from the kernel via the system call interface.

The kernel itself is structured into a number of subsystems which are explained in detail in the high-level design of the TOE. Those are:

- **File and I/O Subsystem**
  Implements all file system object related functions. Functions include those that allow a process to create, maintain, interact and delete file-system objects, such as regular files, directories, symbolic links, hard links, device special files, named pipes, and sockets.
- **Process Subsystem**
  Implements functions related to process and thread management. Functions include those that allow the creation, scheduling, execution, and deletion of process and thread subjects.
- **Memory Subsystem**
  Implements functions related to the management of a system's memory resources. Functions include those that create and manage virtual memory, including management of page tables and paging algorithms.
- **Networking Subsystem**
  This subsystem implements UNIX and internet domain sockets as well as algorithms for scheduling network packets.
- **IPC Subsystem**
  Implements functions related to inter-process communication mechanisms. Functions include those that facilitate controlled sharing of information between processes, allowing them to share data and synchronize their execution in order to interact with a common resource.
- **Audit Subsystem**
  Implements the kernel functions required to intercept system calls and audit them in accordance with the auditing policy defined by the system administrator.
- **Kernel Modules Subsystem**
  This subsystem implements an infrastructure to support loadable modules. Functions include those that load and unload kernel modules.
- **Device Driver Subsystem**
  Implements support for various hardware devices through common, device independent interface.
- **KVM**
  The KVM subsystem provides the kernel parts of the virtualization.
- **Kernel SELinux Subsystem**
  This subsystem provides a framework for various access control policies. The TOE configuration utilizes this subsystem to implement separation of virtual machines.

The trusted processes include the following subsystems:

- **Identification and Authentication**
  This subsystem includes all the processes that are require to identify and authenticate users. All those processes share a common set of functions (pluggable authentication modules (PAM)) that ensure that the same policy will be enforced with respect to identification and authentication of users. Successful as well as unsuccessful authentication attempts can be audited.
- **Network Applications**
  This subsystem includes the trusted processes implementing networking functions. The TOE supports SSH. The secure configuration as defined in the Security Target [6] restricts the cipher suites that can be used for secure communication.
- **System Management**
  This subsystem includes the trusted commands a system administrator can use to manage users and groups, set the time and date and check the integrity of the installed packages.
- **Batch Processing**
  This subsystem includes the cron and at trusted processes that allow to execute user programs at predefined time schedules. They ensure that the users are restricted to the same security policy restrictions that also apply when they start programs interactively.
- **User Level Audit**
  This subsystem includes all the trusted processes and commands outside of the kernel required to collect, store and process audit records.

In addition to those functions the TOE includes a secure system initialization function which brings the TOE into a secure state after it is powered on or after a reset. This function ensures that user interaction with the TOE can only occur after the TOE is securely initialized and in a secure state.

The TOE provides the following security functionality:

- **Identification and Authentication**
  The TOE provides identification and authentication using pluggable authentication modules (PAM) based upon user passwords. The quality of the passwords used can be enforced through configuration options controlled by the TOE. Other authentication methods (e.g. Kerberos authentication, token based authentication) that are supported by the TOE as pluggable authentication modules are not part of the evaluated configuration. Functions that ensure a basic password strength and limit the use of the su command and restrict root login to specific terminals are also included.
- **Audit**
  The TOE provides the capability to audit a large number of events including individual system calls as well as events generated by trusted processes. Audit data is collected in regular files in ASCII format. The TOE provides a program for the purpose of searching the audit records.
  The system administrator can define a rule base to restrict auditing to the events he is interested in. This includes the ability to restrict auditing to specific events, specific users,specific objects or a combination of all of this.
- **Discretionary Access Control**
  Discretionary Access Control (DAC) restricts access to file system objects based on

Access Control Lists (ACLs) that include the standard UNIX permissions for user, group and others. Access control mechanisms also protect IPC objects from unauthorized access. The TOE includes the ext3 file system, which supports POSIX ACLs. This allows defining access rights to files within this type of file system down to the granularity of a single user.

- **Mandatory Access Control**
  Mandatory Access Control (MAC) restricts access to objects based on labels assigned to subjects and objects. The TOE implements non-hierarchical categories to control access to virtual machines.
- **Object Reuse**
  File system objects as well as memory and IPC objects will be cleared before they can be reused by a process belonging to a different user.
- **Security Management**
  The management of the security critical parameters of the TOE is performed by administrative users. A set of commands that require root privileges are used for system management. Security parameters are stored in specific files that are protected by the access control mechanisms of the TOE against unauthorized access by users that are not administrative users.
- **Secure Communication**
  The TOE supports the definition of trusted channels using SSH. Password based authentication is supported. Only a restricted number of cipher suites are supported for those protocols in the evaluated configuration. They are listed in the Security Target [6].
- **TSF Protection**
  While in operation, the kernel software and data are protected by the hardware memory protection mechanisms. The memory and process management components of the kernel ensure a user process cannot access kernel storage or storage belonging to other processes.
  Non-kernel TSF software and data are protected by DAC and process isolation mechanisms. In the evaluated configuration, the reserved user ID root owns the directories and files that define the TSF configuration. In general, files and directories containing internal TSF data (e.g., configuration files, batch job queues) are also protected from reading by DAC permissions.
  The TOE and the hardware and firmware components are required to be physically protected from unauthorized access. The system kernel mediates all access to the hardware mechanisms themselves, other than program visible CPU instruction functions.

6 Documentation

The evaluated documentation as outlined in table 2 is being provided with the product to the customer. This documentation contains the required information for secure usage of the TOE in accordance with the Security Target [6].

Additional obligations and notes for secure usage of the TOE as outlined in chapter 10 of this report have to be followed.

7 IT Product Testing

7.1 Developer Testing

The evaluator examined the information provided by the sponsor and determined the following:

7.1.1 Test configuration

The test results provided by the sponsor were generated on the following systems:

- IBM BladeCenter HS22 64-bit & 32bit
- IBM System x iDataPlex dx360 M2 64-bit & 32bit

The sponsor has performed his tests on the above listed hardware platform. The software was installed and configured as defined in the Evaluated Configuration Guide [8] with additional software packages identified in the Test Plan. The test plan presents the arguments that those additional packages are within the boundary defined by the Security Target and do not constitute a violation of the evaluated configuration.

7.1.2 Testing approach

The test plan provided by the sponsor lists test cases by groups, which reflects the mix of sources for the test cases. The provided mapping lists the SFRs and the TSFI the test cases are associated with. The test plan is focused on the security functions of the TOE and ignores other aspects typically found in developer test plans. The test cases are mapped to the corresponding functional specification and HLD.

The sponsor uses several test suites. The test suites are a mix of automated and manual tests.

The test suite has a common framework for the automated tests in which individual test cases adhere to a common structure for setup, execution and clean up of tests. Each test case may contain several tests of the same function, stressing different parts (for example, base functionality, behavior with illegal parameters and reaction to missing privileges). Each test within a test case reports PASS, OK or FAIL and the test case summary in batch mode reports PASS if all the tests within the test case passed, otherwise FAIL.

The manual tests cover functionality that cannot easily be tested in an automated way, such as console login. Template text files are provided that detail the exact steps required, along with the expected results. The tester creates a copy of the template, inserts the actual results, and compares them with the expected ones manually.

All the tests were executed successfully (pass/ok) apart from the test cases that are documented to fail. The test systems were configured according to the ST and the instructions in [8]. The manual test results also include PASS/FAIL labeling by the sponsor.

7.1.3 Testing results

The test results provided by the sponsor were generated on the hardware platform listed above. As described in the testing approach, the test results of all the automated tests are written to files. The test results of the few manual tests have been recorded by the sponsor and those results have been presented in separate files.

All test results from all tested environments show that the expected test results are identical to the actual test results, considering the expected failures stated in the test plan.

7.1.4 Test coverage

The functional specification has identified the following different TSFI:

- system calls
- security critical configuration files (TSF databases)
- trusted programs and the corresponding network protocol SSH v2.

The mapping provided by the sponsor shows that the tests cover all individual TSFI identified for the TOE. An extension to this mapping developed by the evaluator as documented in the test case coverage analysis document shows that also significant details of the TSFI have been tested with the sponsor's test suite.

7.1.5 Test depth

In addition to the mapping to the functional specification, the developer provided a mapping of test cases to subsystems of the high-level design and the internal interfaces described in the high-level design. This mapping shows that all subsystems and the internal interfaces are covered by test cases. To show evidence that the internal interfaces have been called, the developer provided the description of the internal interfaces as part of the high-level design. The interfaces are clear enough to allow the evaluator to assess whether they have been covered by testing.

Not all of the internal interfaces mentioned in the high-level design could be covered by direct test cases. Some internal interfaces can – due to the restrictions of the evaluated configuration– only be invoked during system startup. This includes especially internal interfaces to load and unload kernel modules, to register / deregister device drivers and install / deinstall interrupt handlers. Since the evaluated configuration does not allow to dynamically load and unload device drivers as kernel modules, those interfaces are only used during system startup and are, therefore, implicitly tested there.

7.1.6 Conclusion

The evaluator has verified that developer testing was performed on hardware conformant to the ST [6].

The evaluator was able to follow and fully understand the developer testing approach by using the information provided by the sponsor.

The evaluator analyzed the developer testing coverage and the depth of the testing by reviewing all test cases. The evaluator found the testing of the TSF to be extensive and covering the TSFI as identified in the functional specification as well as the subsystem/internal interfaces identified in the HLD.

The evaluator reviewed the test results provided by the sponsor and found them to be consistent with the expected test results according to the test plan.

7.2 Evaluator Testing Effort

When performing independent evaluator tests, the evaluator determined the following:

7.2.1 TOE test configuration

The evaluator verified the test systems according to the documentation in the Evaluated Configuration Guide [8] and the test plan. As assessed in the evaluation report on the administrator guidance, [8] is consistent with the ST [6]. Hence, the evaluator concludes that the evaluator's configuration is consistent with the ST [6].

7.2.2 Evaluator tests performed

In addition to participating in all the automated developer tests, the evaluator devised tests for a subset of the TOE. The tests are listed in the Evaluator Test Plan.

The evaluator has chosen these tests for the following reasons:

- The test cases examine some of the security functions of the TOE in more detail than the sponsor supplied test cases. (Object reuse and DAC).
- The test cases cover aspects not included in the developer testing (verification of the ACL support in the archival tool, the use of /dev/random instead of /dev/urandom).
- As the sponsor-supplied test cases already cover the TOE in a broad sense the evaluator has devised only a small set of test cases.

The evaluator created several test cases for testing a few functional aspects where the sponsor test cases were not considered by the evaluator to be broad enough. During the evaluator's review of the test cases provided by the sponsor, the evaluator gained confidence in the sponsor testing effort and the depth of test coverage in the sponsor supplied test cases. The analysis has shown a very wide coverage of the TSF, therefore the evaluator devised only a small number of test cases.

7.2.3 Summary of Evaluator test results

The evaluator testing effort consists of two parts. The first one is the observation of the developer test execution and the second is the execution of the tests created by the evaluator.

The tests were performed remotely at the developer's data center. The systems available for testing are listed above.

In each case the system was accessible through SSH. The TOE operating system with the required additional RPMs as well as the test cases and test tools were installed on the test machine by the developer according to the instructions in [8] and verified by the evaluator. During the evaluation, the file system type ext3 was used for hard disk partitions on the test system. The configuration scripts triggered by the kick start installation ensured the evaluation-compliant system configuration. After running the automated configuration, no further system configuration was performed and only the tools required for testing have been installed. The test systems were therefore configured according to the ST [6] and the instructions in the [8]. The evaluator watched the sponsor during the execution of the test cases. The log files generated by the test cases were analysed for completeness and failures. The sponsor provided automated test cases.

All the test results conformed to the expected test results from the test plan.

In addition to observing the tests that were provided by the developer according to the test plan from the developer, the evaluator decided to run some additional test cases on the provided test systems:

- Permission settings of relevant configuration files
- Verification of the use of SHA512 passwords
- Verification that SSH uses /dev/random instead of /dev/urandom
- Verification that SUID programs do not change the real UID
- Testing of object reuse in regular file system objects
- Check for data import / export with DAC enforcement
- Verification that the permission check during open() is enforced during read() and write()
- Verification of cleaning of environment for SUID/SGID binaries

All tests passed successfully.

7.3 Evaluator Penetration Testing

The evaluator took the following approach to derive penetration tests for the TOE: First the evaluator checked common sources for vulnerabilities of the Linux operating system in general and the TOE in particular to determine:

- Whether the reported vulnerability would affect the evaluated configuration of the TOE in its intended environment. If yes, the evaluator performed a vulnerability analysis.
- If the reported vulnerability has already been fixed in the evaluated configuration of the TOE.

Beside those vulnerabilities reported in common sources the evaluator checked the other evaluation reports for potential vulnerabilities mentioned there. None where identified.

The evaluator decided to not generate simple penetration tests, but instead to for some of the identified potential vulnerabilities perform a source code analysis far deeper than usually done for this evaluation level. The following reasons apply:

- The TOE as an Open Source product is checked for obvious vulnerabilities quite extensively by the Open Source community making the development of high level, simple penetration tests a rather useless task.
- The TOE as an Open Source product is delivered with the full source code, thus allowing the evaluator to perform an analysis to a depth usually not possible for products evaluated at this level. In general, the evaluator believes that a vulnerability analysis based on source code audit is far more accurate than a test case. Per nature, a perceived vulnerability is usually obscure in nature and therefore only exploitable when meeting certain constraints. As the testing may not meet all constraints, a test case indicating that there is no vulnerability does not demonstrate that no vulnerability is present.
- As the source code is publicly available, the evaluator has to assume that potential attackers use the source code to search for potential attack vectors.

The evaluator has performed his analysis on the TOE source code that was installed from the developer distributed source code DVD.

The penetration testing addressed the following security functions:

- TSF Protection

- VM separation

No residual vulnerabilities for the TOE that are exploitable with the assumed attack potential stated in the ST where identified.

8 Evaluated Configuration

This certification covers the following configurations of the TOE: It is based on Red Hat Enterprise Linux 5.6 (RHEL 5.6) with additional packages as listed in table 2. To software is to be used on the following hardware platforms specified in the Security Target [6]:

- IBM System x based on x86 64bit Intel Xeon processors: x3400 M2, x3400 M3, x3500M2, x3500 M3,x3550 M2, x3550 M3, x3620 M3, x3630 M3, x3650 M2, x3650 M3
- IBM BladeCenter: HS22 and HS22V
- IBM iDataPlex: dx360 M2, dx360 M3

The Evaluated Configuration Guide [8] specifies a number of constraints, such as configuration values for various configuration files, specific steps to be taken during installation and information to administrators on how to manage the TOE.

9 Results of the Evaluation

9.1 CC specific results

The Evaluation Technical Report (ETR) [7] was provided by the ITSEF according to the Common Criteria [1], the Methodology [2], the requirements of the Scheme [3] and all interpretations and guidelines of the Scheme (AIS) [4] as relevant for the TOE.

The Evaluation Methodology CEM [2] was used.

As a result of the evaluation the verdict PASS is confirmed for the following assurance components:

- All components of the EAL 4 package including the class ASE as defined in the CC (see also part C of this report)
- The components ALC_FLR.3 augmented for this TOE evaluation.

The evaluation has confirmed:

- for the Functionality: Product specific Security Target
  Common Criteria Part 2 extended
- for the Assurance: Common Criteria Part 3 conformant
  EAL 4 augmented by ALC_FLR.3

The results of the evaluation are only applicable to the TOE as defined in chapter 2 and the configuration as outlined in chapter 8 above.

9.2 Results of cryptographic assessment

The vulnerability assessment results as stated within this certificate do not include a rating for those cryptographic algorithms and their implementation suitable for encryption and decryption (see BSIG Section 9, Para. 4, Clause 2). This holds for: Cryptographically secured communication channels.

The following cryptographic algorithms are used by the TOE to enforce its security policy:

Algorithm	Keyl ength	Intended purpose	Security function	Implementation standard
AES	128 bits, 192 bits 256 bits	encryption / decryption	Identification & Authentication, Protected Data Transfer	RFC 4253
TDES	168 bits	encryption / decryption	Identification & Authentication, Protected Data Transfer	RFC 4253
RSA	1024 bits, 2048 bits, 3072 bits	key generation	Identification & Authentication, Protected Data Transfer	U.S. NIST FIPS PUB 186-3
DSA	L=1024 N=160 bits	key generation	Identification & Authentication, Protected Data Transfer	U.S. NIST FIPS PUB 186-3

Table 3: Cryptographic Functions

The strength of the cryptographic algorithms was not rated in the course of this certification procedure (see BSIG Section 9, Para. 4, Clause 2).

10 Obligations and Notes for the Usage of the TOE

The documents as outlined in table 2 contain necessary information about the usage of the TOE and all security hints therein have to be considered. In addition all aspects of Assumptions, threats and policies as outlined in the Security Target not covered by the TOE itself need to be fulfilled by the operational environment of the TOE.

The customer or user of the TOE shall consider the results of the certification within his system risk management process. In order for the evolution of attack methods and techniques to be covered, he should define the period of time until a re-assessment for the TOE is required and thus requested from the sponsor of the certificate.

If available, certified updates of the TOE should be used. If non-certified updates or patches are available the user of the TOE should request the sponsor to provide a re-certification. In the meantime a risk management process of the system using the TOE should investigate and decide on the usage of not yet certified updates and patches or take additional measures in order to maintain system security.

11 Security Target

For the purpose of publishing, the Security Target [6] of the Target of Evaluation (TOE) is provided within a separate document as Annex A of this report.

12 Definitions

12.1 Acronyms

AIS Application Notes and Interpretations of the Scheme

BSI	Bundesamt für Sicherheit in der Informationstechnik / Federal Office for Information Security, Bonn, Germany
BSIG	BSI-Gesetz / Act on the Federal Office for Information Security
CCRA	Common Criteria Recognition Arrangement
CC	Common Criteria for IT Security Evaluation
CEM	Common Methodology for Information Technology Security Evaluation
EAL	Evaluation Assurance Level
ETR	Evaluation Technical Report
IT	Information Technology
ITSEC	Information Technology Security Evaluation Criteria
ITSEF	Information Technology Security Evaluation Facility
PP	Protection Profile
RHEL	Red Hat Enterprise Linux
SAR	Security Assurance Requirement
SFP	Security Function Policy
SFR	Security Functional Requirement
ST	Security Target
TOE	Target of Evaluation
TSF	TOE Security Functionalities

12.2 Glossary

Augmentation - The addition of one or more requirement(s) to a package.

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.

Formal - Expressed in a restricted syntax language with defined semantics based on well-established mathematical concepts.

Informal - Expressed in natural language.

Object - An passive entity in the TOE, that contains or receives information, and upon which subjects perform operations.

Protection Profile - An implementation-independent statement of security needs for a TOE type.

Security Target - An implementation-dependent statement of security needs for a specific identified TOE.

Subject - An active entity in the TOE that performs operations on objects.

Target of Evaluation - A set of software, firmware and/or hardware possibly accompanied by guidance.

TOE Security Functionality - combined functionality of all hardware, software, and firmware of a TOE that must be relied upon for the correct enforcement of the SFRs

13 Bibliography

[1] Common Criteria for Information Technology Security Evaluation, Version 3.1,
Part 1: Introduction and general model, Revision 3, July 2009
Part 2: Security functional components, Revision 3, July 2009
Part 3: Security assurance components, Revision 3, July 2009

[2] Common Methodology for Information Technology Security Evaluation (CEM), Evaluation Methodology, Version 3.1, Rev. 3, July 2009

[3] BSI certification: Procedural Description (BSI 7125)

[4] Application Notes and Interpretations of the Scheme (AIS) as relevant for the TOE[8].

[5] German IT Security Certificates (BSI 7148), periodically updated list published also in the BSI Website

[6] Security Target BSI-DSZ-CC-0724-2012, Version 0.12, 2011-05-18, Red Hat Enterprise Linux 5.6 KVM Security Target, Red Hat, Inc.

[7] Evaluation Technical Report, Version 4, 2012-03-30, Final Evaluation Technical Report, atsec information security GmbH, (confidential document)

[8] EAL4 Evaluated Configuration Guide for Red Hat Enterprise Linux with KVM support on IBM hardware, Version 1.4, 2011-12-14

[8] specifically

- AIS 20, Version 1, 2. December 1999, Funktionalitätsklassen und Evaluationsmethodologie für deterministische Zufallszahlengeneratoren
- AIS 32, Version 6, 3 August 2010, CC-Interpretationen im deutschen Zertifizierungsschema

C Excerpts from the Criteria

CC Part1:

Conformance Claim (chapter 10.4)

“The conformance claim indicates the source of the collection of requirements that is met by a PP or ST that passes its evaluation. This conformance claim contains a CC conformance claim that:

- describes the version of the CC to which the PP or ST claims conformance.
- describes the conformance to CC Part 2 (security functional requirements) as either:
  - **CC Part 2 conformant** - A PP or ST is CC Part 2 conformant if all SFRs in that PP or ST are based only upon functional components in CC Part 2, or
  - **CC Part 2 extended** - A PP or ST is CC Part 2 extended if at least one SFR in that PP or ST is not based upon functional components in CC Part 2.
- describes the conformance to CC Part 3 (security assurance requirements) as either:
  - **CC Part 3 conformant** - A PP or ST is CC Part 3 conformant if all SARs in that PP or ST are based only upon assurance components in CC Part 3, or
  - CC Part 3 extended - A PP or ST is CC Part 3 extended if at least one SAR in that PP or ST is not based upon assurance components in CC Part 3.

Additionally, the conformance claim may include a statement made with respect to packages, in which case it consists of one of the following:

- Package name Conformant - A PP or ST is conformant to a pre-defined package (e.g. EAL) if:
  - the SFRs of that PP or ST are identical to the SFRs in the package, or
  - the SARs of that PP or ST are identical to the SARs in the package.
- Package name Augmented - A PP or ST is an augmentation of a predefined package if:
  - the SFRs of that PP or ST contain all SFRs in the package, but have at least one additional SFR or one SFR that is hierarchically higher than an SFR in the package.
  - the SARs of that PP or ST contain all SARs in the package, but have at least one additional SAR or one SAR that is hierarchically higher than an SAR in the package.

Note that when a TOE is successfully evaluated to a given ST, any conformance claims of the ST also hold for the TOE. A TOE can therefore also be e.g. CC Part 2 conformant.

Finally, the conformance claim may also include two statements with respect to Protection Profiles:

- PP Conformant - A PP or TOE meets specific PP(s), which are listed as part of the conformance result.
- Conformance Statement (Only for PPs) - This statement describes the manner in which PPs or STs must conform to this PP: strict or demonstrable. For more information on this Conformance Statement, see Annex D.”

CC Part 3:

Class APE: Protection Profile evaluation (chapter 10)

“Evaluating a PP is required to demonstrate that the PP is sound and internally consistent, and, if the PP is based on one or more other PPs or on packages, that the PP is a correct instantiation of these PPs and packages. These properties are necessary for the PP to be suitable for use as the basis for writing an ST or another PP.

Assurance Class	Assurance Components
Class APE: Protection Profile evaluation	APE_INT.1 PP introduction
	APE_CCL.1 Conformance claims
	APE_SPD.1 Security problem definition
	APE_OBJ.1 Security objectives for the operational environment APE_OBJ.2 Security objectives
	APE_ECD.1 Extended components definition
	APE_REQ.1 Stated security requirements APE_REQ.2 Derived security requirements

APE: Protection Profile evaluation class decomposition”

Class ASE: Security Target evaluation (chapter 11)

“Evaluating an ST is required to demonstrate that the ST is sound and internally consistent, and, if the ST is based on one or more PPs or packages, that the ST is a correct instantiation of these PPs and packages. These properties are necessary for the ST to be suitable for use as the basis for a TOE evaluation.”

Assurance Class	Assurance Components
Class ASE: Security Target evaluation	ASE_INT.1 ST introduction
	ASE_CCL.1 Conformance claims
	ASE_SPD.1 Security problem definition
	ASE_OBJ.1 Security objectives for the operational environment ASE_OBJ.2 Security objectives
	ASE_ECD.1 Extended components definition
	ASE_REQ.1 Stated security requirements ASE_REQ.2 Derived security requirements
	ASE_TSS.1 TOE summary specification ASE_TSS.2 TOE summary specification with architectural design summary

ASE: Security Target evaluation class decomposition

Security assurance components (chapter 7)

“The following Sections describe the constructs used in representing the assurance classes, families, and components.“
“Each assurance class contains at least one assurance family.”
“Each assurance family contains one or more assurance components.”

The following table shows the assurance class decomposition.

Assurance Class	Assurance Components
ADV: Development	ADV_ARC.1 Security architecture description
	ADV_FSP.1 Basic functional specification ADV_FSP.2 Security-enforcing functional specification ADV_FSP.3 Functional specification with complete summary ADV_FSP.4 Complete functional specification ADV_FSP.5 Complete semi-formal functional specification with additional error information ADV_FSP.6 Complete semi-formal functional specification with additional formal specification
	ADV_IMP.1 Implementation representation of the TSF ADV_IMP.2 Implementation of the TSF
	ADV_INT.1 Well-structured subset of TSF internals ADV_INT.2 Well-structured internals ADV_INT.3 Minimally complex internals
	ADV_SPM.1 Formal TOE security policy model
	ADV_TDS.1 Basic design ADV_TDS.2 Architectural design ADV_TDS.3 Basic modular design ADV_TDS.4 Semiformal modular design ADV_TDS.5 Complete semiformal modular design ADV_TDS.6 Complete semiformal modular design with formal high-level design presentation

Assurance Class	Assurance Components
AGD: Guidance documents	AGD_OPE.1 Operational user guidance
	AGD_PRE.1 Preparative procedures
ALC: Life cycle support	ALC_CMC.1 Labelling of the TOE ALC_CMC.2 Use of a CM system ALC_CMC.3 Authorisation controls ALC_CMC.4 Production support, acceptance procedures and automation ALC_CMC.5 Advanced support
	ALC_CMS.1 TOE CM coverage ALC_CMS.2 Parts of the TOE CM coverage ALC_CMS.3 Implementation representation CM coverage ALC_CMS.4 Problem tracking CM coverage ALC_CMS.5 Development tools CM coverage
	ALC_DEL.1 Delivery procedures
	ALC_DVS.1 Identification of security measures ALC_DVS.2 Sufficiency of security measures
	ALC_FLR.1 Basic flaw remediation ALC_FLR.2 Flaw reporting procedures ALC_FLR.3 Systematic flaw remediation
	ALC_LCD.1 Developer defined life-cycle model ALC_LCD.2 Measurable life-cycle model
	ALC_TAT.1 Well-defined development tools ALC_TAT.2 Compliance with implementation standards ALC_TAT.3 Compliance with implementation standards - all parts
ATE: Tests	ATE_COV.1 Evidence of coverage ATE_COV.2 Analysis of coverage ATE_COV.3 Rigorous analysis of coverage
	ATE_DPT.1 Testing: basic design ATE_DPT.2 Testing: security enforcing modules ATE_DPT.3 Testing: modular design ATE_DPT.4 Testing: implementation representation
	ATE_FUN.1 Functional testing ATE_FUN.2 Ordered functional testing
	ATE_IND.1 Independent testing – conformance ATE_IND.2 Independent testing – sample ATE_IND.3 Independent testing – complete
AVA: Vulnerability assessment	AVA_VAN.1 Vulnerability survey AVA_VAN.2 Vulnerability analysis AVA_VAN.3 Focused vulnerability analysis AVA_VAN.4 Methodical vulnerability analysis AVA_VAN.5 Advanced methodical vulnerability analysis

Assurance class decomposition

Evaluation assurance levels (chapter 8)

“The Evaluation Assurance Levels (EALs) provide an increasing scale that balances the level of assurance obtained with the cost and feasibility of acquiring that degree of assurance. The CC approach identifies the separate concepts of assurance in a TOE at the end of the evaluation, and of maintenance of that assurance during the operational use of the TOE.

It is important to note that not all families and components from CC Part 3 are included in the EALs. This is not to say that these do not provide meaningful and desirable assurances. Instead, it is expected that these families and components will be considered for augmentation of an EAL in those PPs and STs for which they provide utility.”

Evaluation assurance level (EAL) overview (chapter 8.1)

“Table 1 represents a summary of the EALs. The columns represent a hierarchically ordered set of EALs, while the rows represent assurance families. Each number in the resulting matrix identifies a specific assurance component where applicable.

As outlined in the next Section, seven hierarchically ordered evaluation assurance levels are defined in the CC for the rating of a TOE's assurance. They are hierarchically ordered inasmuch as each EAL represents more assurance than all lower EALs. The increase in assurance from EAL to EAL is accomplished by substitution of a hierarchically higher assurance component from the same assurance family (i.e. increasing rigour, scope, and/or depth) and from the addition of assurance components from other assurance families (i.e. adding new requirements).

These EALs consist of an appropriate combination of assurance components as described in Chapter 7 of this CC Part 3. More precisely, each EAL includes no more than one component of each assurance family and all assurance dependencies of every component are addressed.

While the EALs are defined in the CC, it is possible to represent other combinations of assurance. Specifically, the notion of “augmentation” allows the addition of assurance components (from assurance families not already included in the EAL) or the substitution of assurance components (with another hierarchically higher assurance component in the same assurance family) to an EAL. Of the assurance constructs defined in the CC, only EALs may be augmented. The notion of an “EAL minus a constituent assurance component” is not recognised by the standard as a valid claim. Augmentation carries with it the obligation on the part of the claimant to justify the utility and added value of the added assurance component to the EAL. An EAL may also be augmented with extended assurance requirements.

Assurance Class	Assurance Family	Assurance Components by Evaluation Assurance Level						
		EAL1	EAL2	EAL3	EAL4	EAL5	EAL6	EAL7
Development	ADV_ARC		1	1	1	1	1	1
	ADV_FSP	1	2	3	4	5	5	6
	ADV_IMP				1	1	2	2
	ADV_INT					2	3	3
	ADV_SPM						1	1
	ADV_TDS		1	2	3	4	5	6
Guidance Documents	AGD_OPE	1	1	1	1	1	1	1
	AGD_PRE	1	1	1	1	1	1	1
Life cycle Support	ALC_CMC	1	2	3	4	4	5	5
	ALC_CMS	1	2	3	4	5	5	5
	ALC_DEL		1	1	1	1	1	1
	ALC_DVS			1	1	1	2	2
	ALC_FLR							
	ALC_LCD			1	1	1	1	2
	ALC_TAT				1	2	3	3
Security Target Evaluation	ASE_CCL	1	1	1	1	1	1	1
	ASE_ECD	1	1	1	1	1	1	1
	ASE_INT	1	1	1	1	1	1	1
	ASE_OBJ	1	2	2	2	2	2	2
	ASR_REQ	1	2	2	2	2	2	2
	ASE_SPD		1	1	1	1	1	1
	ASE_TSS	1	1	1	1	1	1	1
Tests	ATE_COV		1	2	2	2	3	3
	ATE_DPT			1	1	3	3	4
	ATE_FUN		1	1	1	1	2	2
	ATE_IND	1	2	2	2	2	2	3
Vulnerability assessment	AVA_VAN	1	2	2	3	4	5	5

Table 1: Evaluation assurance level summary”

Evaluation assurance level 1 (EAL1) - functionally tested (chapter 8.3)

“Objectives

EAL1 is applicable where some confidence in correct operation is required, but the threats to security are not viewed as serious. It will be of value where independent assurance is required to support the contention that due care has been exercised with respect to the protection of personal or similar information.

EAL1 requires only a limited security target. It is sufficient to simply state the SFRs that the TOE must meet, rather than deriving them from threats, OSPs and assumptions through security objectives.

EAL1 provides an evaluation of the TOE as made available to the customer, including independent testing against a specification, and an examination of the guidance documentation provided. It is intended that an EAL1 evaluation could be successfully conducted without assistance from the developer of the TOE, and for minimal outlay.

An evaluation at this level should provide evidence that the TOE functions in a manner consistent with its documentation.”

Evaluation assurance level 2 (EAL2) - structurally tested (chapter 8.4)

“Objectives

EAL2 requires the co-operation of the developer in terms of the delivery of design information and test results, but should not demand more effort on the part of the developer than is consistent with good commercial practise. As such it should not require a substantially increased investment of cost or time.

EAL2 is therefore applicable in those circumstances where developers or users require a low to moderate level of independently assured security in the absence of ready availability of the complete development record. Such a situation may arise when securing legacy systems, or where access to the developer may be limited.”

Evaluation assurance level 3 (EAL3) - methodically tested and checked (chapter 8.5)

“Objectives

EAL3 permits a conscientious developer to gain maximum assurance from positive security engineering at the design stage without substantial alteration of existing sound development practises.

EAL3 is applicable in those circumstances where developers or users require a moderate level of independently assured security, and require a thorough investigation of the TOE and its development without substantial re-engineering.”

Evaluation assurance level 4 (EAL4) - methodically designed, tested, and reviewed (chapter 8.6)

“Objectives

EAL4 permits a developer to gain maximum assurance from positive security engineering based on good commercial development practises which, though rigorous, do not require substantial specialist knowledge, skills, and other resources. EAL4 is the highest level at which it is likely to be economically feasible to retrofit to an existing product line.

EAL4 is therefore applicable in those circumstances where developers or users require a moderate to high level of independently assured security in conventional commodity TOEs and are prepared to incur additional security-specific engineering costs.”

Evaluation assurance level 5 (EAL5) - semiformally designed and tested (chapter 8.7)

“Objectives

EAL5 permits a developer to gain maximum assurance from security engineering based upon rigorous commercial development practises supported by moderate application of specialist security engineering techniques. Such a TOE will probably be designed and developed with the intent of achieving EAL5 assurance. It is likely that the additional costs attributable to the EAL5 requirements, relative to rigorous development without the application of specialised techniques, will not be large.

EAL5 is therefore applicable in those circumstances where developers or users require a high level of independently assured security in a planned development and require a rigorous development approach without incurring unreasonable costs attributable to specialist security engineering techniques.”

Evaluation assurance level 6 (EAL6) - semiformally verified design and tested (chapter 8.8)

“Objectives

EAL6 permits developers to gain high assurance from application of security engineering techniques to a rigorous development environment in order to produce a premium TOE for protecting high value assets against significant risks.

EAL6 is therefore applicable to the development of security TOEs for application in high risk situations where the value of the protected assets justifies the additional costs.”

Evaluation assurance level 7 (EAL7) - formally verified design and tested (chapter 8.9)

“Objectives

EAL7 is applicable to the development of security TOEs for application in extremely high risk situations and/or where the high value of the assets justifies the higher costs. Practical application of EAL7 is currently limited to TOEs with tightly focused security functionality that is amenable to extensive formal analysis.”

Class AVA: Vulnerability assessment (chapter 16)

“The AVA: Vulnerability assessment class addresses the possibility of exploitable vulnerabilities introduced in the development or the operation of the TOE.”

Vulnerability analysis (AVA_VAN) (chapter 16.1)

"Objectives

Vulnerability analysis is an assessment to determine whether potential vulnerabilities identified, during the evaluation of the development and anticipated operation of the TOE or by other methods (e.g. by flaw hypotheses or quantitative or statistical analysis of the security behaviour of the underlying security mechanisms), could allow attackers to violate the SFRs.

Vulnerability analysis deals with the threats that an attacker will be able to discover flaws that will allow unauthorised access to data and functionality, allow the ability to interfere with or alter the TSF, or interfere with the authorised capabilities of other users.”

This page is intentionally left blank

D Annexes

List of annexes of this certification report

Annex A: Security Target provided within a separate document.

This page is intentionally left blank.