Federal Office
for Information Security

Certification Report

BSI-DSZ-CC-0924-2014

for

Red Hat Enterprise Linux on 32 bit x86 Architecture, Version 6.2

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-0924-2014

Operating System

Red Hat Enterprise Linux on 32 bit x86 Architecture
Version 6.2

from Red Hat, Inc.

SOGIS
Recognition Agreement

PP Conformance:	Operating System Protection Profile, Version 2.0, 01 June 2010, BSI-CC-PP-0067-2010, OSPP Extended Package – Advanced Audit, Version 2.0, 28 May 2010
Functionality:	PP conformant Common Criteria Part 2 extended
Assurance:	Common Criteria Part 3 conformant EAL 4 augmented by ALC_FLR.3

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. CC and CEM are also published as ISO/IEC 15408 and ISO/IEC 18045.

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.

Common Criteria
Recognition Arrangement

Bonn, 27 November 2014
For the Federal Office for Information Security

Joachim Weber
Head of Division

L.S.

DAkkS
Deutsche
Akkreditierungsstelle
D-ZE-19615-01-00

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

A Certification

1 Specifications of the Certification Procedure

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

- Act on the Federal Office for Information Security[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 ISO/IEC 17065 standard
- BSI certification: Technical information on the IT security certification, Procedural Description (BSI 7138) [3]
- BSI certification: Requirements regarding the Evaluation Facility (BSI 7125) [3]
- Common Criteria for IT Security Evaluation (CC), Version 3.1[5] [1] also published as ISO/IEC 15408.
- Common Methodology for IT Security Evaluation (CEM), Version 3.1 [2] also published as ISO/IEC 18045.
- 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.

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.

---

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

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.

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.

International Recognition of CC – Certificates (CCRA)

The international arrangement on the mutual recognition of certificates based on the CC (Common Criteria Recognition Arrangement, CCRA-2014) has been ratified on 08 September 2014. It covers CC certificates based on collaborative Protection Profiles (cPP) (exact use), certificates based on assurance components up to and including EAL2 or the assurance family Flaw Remediation (ALC_FLR) and certificates for Protection Profiles and for collaborative Protection Profiles (cPP).

The CCRA-2014 replaces the old CCRA signed in May 2000 (CCRA-2000). Certificates based on CCRA-2000, issued before 08 September 2014 are still under recognition according to the rules of CCRA-2000. For on 08 September 2014 ongoing certification procedures and for Assurance Continuity (maintenance and re-certification) of old certificates a transition period on the recognition of certificates according to the rules of CCRA-2000 (i.e. assurance components up to and including EAL4 or the assurance family Flaw Remediation (ALC_FLR)) is defined until 08 September 2017.

As of September 2014 the signatories of the new CCRA are government representatives from the following nations: Australia, Austria, Canada, Czech Republic, Denmark, Finland, France, Germany, Greece, Hungary, India, Israel, Italy, Japan, Malaysia, The Netherlands, New Zealand, Norway, Pakistan, Republic of Korea, Singapore, Spain, Sweden, Turkey, United Kingdom, and the United States.

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.

As this certificate is a re-certification of a certificate issued according to CCRA-2000 this certificate is recognized according to the rules of CCRA-2000, i.e. for all assurance components selected.

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 on 32 bit x86 Architecture, Version 6.2 has undergone the certification procedure at BSI. This is a re-certification based on BSI-DSZ-CC-0754-2012. Specific results from the evaluation process BSI-DSZ-CC-0754-2012 were re-used.

The evaluation of the product Red Hat Enterprise Linux on 32 bit x86 Architecture, Version 6.2 was conducted by atsec information security GmbH. The evaluation was completed on 19 November 2014. 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,
- the product is operated in the environment described, as 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.

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.

[6] Information Technology Security Evaluation Facility

5 Publication

The product Red Hat Enterprise Linux on 32 bit x86 Architecture, Version 6.2 has been included in the BSI list of 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.

[7] Red Hat, Inc.
Red Hat Tower
100 East Davie Street
Raleigh, NC 27601
USA

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 Target of Evaluation (TOE) is a configurable Linux-based operating system, which has been developed to provide a good level of security as required in commercial environments. It also meets all requirements of the Operating System protection profile [7] together with the Extended Package for Audit. Key security features are: Auditing, encrypted storage and communication, packet filtering, identification and authentication, discretionary access control, security management and TOE self protection.

The Security Target [6] is the basis for this certification. It is based on the certified Protection Profile Operating System Protection Profile, Version 2.0, 01 June 2010, BSI-CC-PP-0067-2010,
OSPP Extended Package – Advanced Audit, Version 2.0, 28 May 2010 [7].

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 6.1. 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 Functionality:

TOE Security Functionality	Addressed issue
Auditing	The Lightweight Audit Framework (LAF) is designed to be an audit system making 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
Cryptographic support	The TOE provides cryptographically secured communication channels as well as cryptographic primitives that unprivileged users can utilize for unspecified purposes. The TOE provides cryptographically secured communication to allow remote entities to log into the TOE. For interactive usage, the SSHv2 protocol is provided.
Packet filter	The TOE provides a stateless and stateful packet filter for regular IP-based communication. Layer 3 (IP) and layer 4 (TCP, UDP, ICMP) network protocols can be controlled using this packet filter. Ethernet frames routed through bridges are controlled by a separate packet filter which implements a stateless packet filter for the TCP/IP protocol family.
Identification and Authentication	User identification and authentication in the TOE includes all forms of interactive login (e.g. using the SSHv2 protocol or log in at the local console) as well as identity changes through the su or sudo command. These all rely on explicit authentication information provided interactively by a user.
Discretionary Access Control (DAC)	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.
Confidentiality-protected data	Using dm_crypt, the Linux operating system offers administrators and

TOE Security Functionality	Addressed issue
storage	users cryptographically protected block device storage space. Only with the passphrase can the session key used for encryption or decryption be obtained and used. Any data stored on the block devices protected by dm_crypt is encrypted and cannot be accessed even when the TOE is not operational unless the TOE is operational and the block device session key is unlocked.
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.
Protection mechanisms	The TOE provides mechanisms to prevent common buffer overflow and similar attacks. These mechanisms are used for the TSF and are available to untrusted code.

Table 1: TOE Security Functionalities

For more details please refer to the Security Target [6], chapter 7.1.

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 configurations of the TOE as outlined in chapter 8.

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 on 32 bit x86 Architecture, Version 6.2

The following table outlines the TOE deliverables:

No	Type	Identifier	Release	Form of Delivery
1	SW	Red Hat Enterprise Linux 6.2 Server, 32 bit x86 Architecture rhel-server-6.2-i386-dvd.iso SHA-256 Checksum: 0c3a57cf048c0f21a45e1be039afe433a2ab6dc26996519f6d63e27932b15d84	6.2	Download
2	SW	Evaluation package RPM EAL4_RHEL6.2 (cc-eal4-config-rhel62-i386-0.2-1.fc20.noarch.rpm)	1.fc20	Download
3	DOC	EAL4 Evaluated Configuration Guide for Red Hat Enterprise Linux RHEL62-EAL4-Configuration-Guide.pdf SHA-256 Checksum: 2f9e0c42e14215a0af8a737234188d34098d1d95533313799f5dc0f4818e900e	v2.10	Download

No	Type	Identifier	Release	Form of Delivery
4	SW	The following patches from EUS 6.2.z: • Fix RHSA-2013:16082 • Fix RHBA-2012:0338 • Fix RHEA-2012:0065 • Fix RHBA-2012:0134 • Fix RHBA-2012:1319-2 • Fix RHBA-2012:0337 • Fix RHBA-2012:0344 • Fix RHBA-2012:0339 • Fix RHSA-2012:0699-01 • Fix RHEA-2012:0486 • Fix RHSA-2012:0451	see package names	Download

Table 2: Deliverables of the TOE

Overview of Delivery Procedure

The TOE is delivered from the developer, Red Hat, using the Red Hat delivery mechanism. There are several download components: the Red Hat Enterprise Linux Server 6.2 32-bit distribution (ISO file), an additional package created specifically for the evaluation of RHEL 6.2 (containing the kickstart file, Evaluated Configuration Guide, and configuration files), and multiple additional packages that must be installed to achieve the TOE. The packages and ISO files are delivered via the same delivery mechanism.

RHEL 6.2 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 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 are advised to verify the checksums and the signatures.

The download is securely provided by the developer, reviewed and built into an RPM, 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.

Identification of the TOE by the User

The customer can identify the TOE packages in the download sites by appropriate labeling. The download page lists the release and the architecture (for example Red Hat Enterprise Linux Server (v. 6.2 for 32-bit i686)). The downloaded ISO image is named according to release and architecture. 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 6.2".

For all packages, the user can verify their integrity by downloading the RedHat signing key from the download website and running the rpm --checksig command as described in the Evaluated Configuration Guide. To verify whether the correct versions of the packages

have been installed, users can use the rpm -qa command and search the output for the respective packages.

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:

- Auditing
- Cryptographically protected Remote Access
- Discrenationary Access
- Network Flow Control
- Sunbject Information Flow Control
- Identification and Authentication
- Management of I&A
- Trusted Channel to Remote IT
- Remote Audit Trail
- Audit Analysis
- Protect the confidentiality of user data
- Management of trust anchor
- Runtime Protection

4 Assumptions and Clarification of Scope

The Assumptions defined in the Security Target 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: competent and trustworthy administrators, trusted remote IT systems, correct configration and setup of system, system maintenance, trusted physical environment, secure recovery mechanisms. Details can be found in the Security Target [6], chapter 4.2.

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, 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. Those configuration files are also protected by the file system discretionary access control security function enforced by the kernel.

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.

- **Cryptographic mechanisms**
  Cryptographic mechanisms implemented in the kernel which can be used as a library for other kernel parts, if needed.

The trusted processes include the following subsystems:

- **Identification and Authentication**
  This subsystem includes all the processes that are required 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 SSHv2. The secure configuration as defined in the Security Target 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 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. Audit records can be transferred to a remote audit daemon.

- **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 ext4 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.
- **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 SSHv2. 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.
- **Storage encryption**
  The TOE supports encrypted block devices to provide storage confidentiality via dm_crypt.
- **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.
  In addition, mechanisms for protection against stack overflow attacks are provided.

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.

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 test results provided by the sponsor were generated on the following systems: Northrop Grumman Payload Control Element (PCE) Server 309-C20213. The software was installed and configured as defined in the Evaluated Configuration Guide [10] with additional software packages identified in the Test Plan (confidential).

7.2 Developer 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 one test suite which pulls in tests from older test suites (Linux Test Project) for some specific cases, but the actual handling of this is transparent to the user. 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 cleanup 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.

All the tests were executed successfully (pass). The test systems were configured according to the ST and the instructions in [10].

7.3 Developer 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.

All test results from all tested environments show that the expected test results are identical to the actual test results.

7.4 Developer 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 SSHv2.

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 adds to this coverage and depth analysis.

7.5 Developer 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 / de-register device drivers and install / de-install 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.

7.6 Conclusion

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

7.7 Evaluator Testing Effort

The evaluator devised tests for a subset of the TOE. The tests are listed in the Evaluator Test Plan. The evaluator chose to focus independent testing on the RELRO security functionality since this is a relatively new functionality. The evaluator also repeated the previous evaluation's test cases after some clean up of the test case source code.

7.8 Summary of Evaluator test results

The evaluator ran the independent test cases on the following platform:

- Northrop Grumman Payload Control Element (PCE) Server 309-C20213

The sponsor provided full and unlimited access to remote tests systems. After successful installation all systems could be accessed via SSHv2. All tests were executed via this laptop.

The system was accessible through SSHv2. 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 [10] and verified by the evaluator. During the evaluation, the file system type ext4 was used for hard disk partitions on the test system. The configuration scripts triggered by the kickstart 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 where installed. The test systems were therefore configured according to the ST [6] and the

instructions in the Guidance documentation [10]. The evaluator verified the configuration against the Evaluated Configuration Guide [10] before conducting the independent tests.

The evaluator ran independent test cases on the provided test systems:

- Fail safe settings for the TSF (RELRO, address randomization, and stack protection)
- Fail safe settings for user programs (RELRO, address randomization, and stack protection) Non-executable stack
- Verification of address space randomization
- Permission settings of relevant configuration files
- Verification of the use of SHA512 passwords
- Verification that SSHv2 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

After some anomalies with RELRO support for libgcc were addressed, all tests passed successfully.

7.9 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 has already been fixed in the evaluated configuration of the TOE. If not, whether the reported vulnerability would affect the evaluated configuration of the TOE in its intended environment. If yes, the evaluator performed a vulnerability analysis.

The evaluator has performed his analysis on the TOE source code that was installed from the developer distributed source code DVD. The analysis addressed the TSF protection of the TOE. The source code analysis did not reveal any vulnerabilities.

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

7.10 Cryptographic Vulnerability Analysis

A cryptographic vulnerability analysis was performed to verify that the employed algorithms are implemented in away that they provide sufficient resistance against attacks. The analysis and testing covered the following topics:

- Statistic analysis of DSA and RSA key generation
  Keys generated via OpenSSL were analyzed for statistical anomalies and border cases.
- Code analysis of dm-crypt
  The dm-crypt methods of key generation and management were examined to verify the quality key generation and protection of the master key.

- Code analysis of OpenSSL RSA
  The RSA key generation was analyzed for appropriate mechanisms to deflect timing attacks.
- Code analysis of OpenSSL DSA
  The DSA key generation was analyzed for appropriate mechanisms to deflect timing attacks.
- Side channel analysis for OpenSSHv2
  The implementation of SSHv2 was analyzed for potential side channels.
- Side channel analysis for dm-crypt
  The implementation of dm-crypt was analyzed for potential side channels.

The implementation of the symmetric algorithms used by the TOE was analyzed for correct implementation and did not reveal any relevant side channels in the intended operational environment.

The analysis did not reveal any residual vulnerabilities.

8 Evaluated Configuration

This certification covers the following configurations of the TOE:

The evaluated configuration is documented in the Evaluated Configuration Guide [10]. It is based on Red Hat Enterprise Linux 6.2 (RHEL 6.2) with additional packages as listed in table 2. The software is to be used on the following hardware platforms specified in the Security Target [6]:

- Northrop Grumman Payload Control Element (PCE) Server 309-C20213

This Evaluated Configuration Guide 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

CC specific results

The Evaluation Technical Report (ETR) [9] 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 and guidance specific for the technology of the product [4] (AIS 34).

For RNG assessment the scheme interpretations AIS 20, AIS 31 was used (see [4]).

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.

As the evaluation work performed for this certification procedure was carried out as a re-evaluation based on the certificate BSI-DSZ-CC-0754-2012, re-use of specific

evaluation tasks was possible. The focus of this re-evaluation was on the use of a different platform: an Intel x86 compatible with 32 bit word size. User space and kernel space are compiled in 32 bit.

The evaluation has confirmed:

- PP Conformance: Operating System Protection Profile, Version 2.0, 01 June 2010, BSI-CC-PP-0067-2010, OSPP Extended Package – Advanced Audit, Version 2.0, 28 May 2010 [7]
- for the Functionality: PP conformant
  Common Criteria Part 2 extended
- for the Assurance: Common Criteria Part 3 conformant
  EAL 4 augmented by ALC_FLR.3

Results of cryptographic assessment

The strength of the cryptographic algorithms was not rated in the course of this certification procedure (see BSIG Section 9, Para. 4, Clause 2). But Cryptographic Functionalities with a security level of lower than 100 bits can no longer be regarded as secure without considering the application context. Therefore, for these functionalities it shall be checked whether the related crypto operations are appropriate for the intended system. Some further hints and guidelines can be derived from the 'Technische Richtlinie BSI TR-02102' (https://www.bsi.bund.de).

Any Cryptographic Functionality that is marked in column '*Security Level above 100 Bits*' of the following table with '*no*' achieves a security level of lower than 100 Bits (in general context).

No.	Purpose	Cryptographic Mechanism	Standard of Implementation	Key Size in Bits	Security Level above 100 Bits	Comments
1	Confidentiality	AES in CBC mode (essiv) using SHA-{1, 224, 256, 384, 512} for IV=sector number (dm-crypt) AES in XTS mode with tweak=sector number (dm-crypt) AES in GCM mode with IV=sector number (dm-crypt) AES in CTR mode with nonce=sector number AES in GCM mode using only encryption with IV=sector number	[FIPS 197] (AES) [SP800-38A] (CBC,CTR) [SP 800-38D] (GCM) [IEEE Std 1619-2007] (XTS) [FIPS180-2] (SHA)	|k|= 128, 192, 256	yes	
2		Serpent in the same modes as AES	(Serpent) [SP800-38A] (CBC,CTR) [SP 800-38D] (GCM) [IEEE Std 1619-2007] (XTS)	|k|= 128, 192, 256	yes	

No.	Purpose	Cryptographic Mechanism	Standard of Implementation	Key Size in Bits	Security Level above 100 Bits	Comments
			[FIPS180-2] (SHA)			
3		Twofish in the same modes as AES	(Twofish) [SP800-38A] (CBC,CTR) [SP 800-38D] (GCM) [IEEE Std 1619-2007] (XTS) [FIPS180-2] (SHA)	|k|= 128, 192, 256	yes	
4	Key Derivation	PBKDF2 based on SHA-1	SP800-132, LUKS	Complex password of at least 8 characters	yes	
5	Authentication	DSA signature generation and verification using SHA-1 (ssh-dss)	[PKCS#1 v2.1] (RSA), [FIPS180-2] (SHA)	L=1024 N=160 bits	no	
6		RSA signature generation and verification (RSASSA-PKCS1-v1_5) using SHA-1 (ssh-rsa)	[PKCS#1 v2.1] (RSA), [FIPS180-2] (SHA)	1024	no	
				2048	no	
				3072	no	
7		Password authentication	n/a	8/12 chars minimum	no	Password rules according to [10]
8	Integrity	HMAC-SHA1 (SSH)	[FIPS 180-2] (SHA) [RFC 2104] (HMAC)	|k|=160	yes	Note: HMAC not shortened in evaluated configuration, see ECG
9	Key Agreement	DH (SSH)	[RFC 4253] (SSH) (SP800-56A) RFC2409 diffie-hellman-group1-sha1	Plength = 1024	no	
10		DH (SSH)	[RFC 4253] (SSH) (SP800-56A) RFC2409 diffie-hellman-group14-sha1	Plength = 2048	yes	
11		DH (SSH)	[RFC 4253] (SSH) (SP800-56A)	plength < 1976	no	Depends on negotiation with client.

No.	Purpose	Cryptographic Mechanism	Standard of Implementation	Key Size in Bits	Security Level above 100 Bits	Comments
			diffie-hellman-group-exchange-sha1 diffie-hellman-group-exchange-sha256	plength ≥ 1976	yes	
12	Confidentiality	TDES in CBC mode (SSH)	[FIPS 46-3] (DES) [SP800-38A] (CBC)	|k|= 168	yes	
13		AES in CBC or CTR mode (SSH)	[FIPS 197] (AES) [SP800-38A] (CBC)	|k|= 128, 192, 256	yes	
14	Trusted Channel	SSH (Version 2.0)	[RFC 4253] (SSH v2.0)	n/a	no	Sec Level = No because of the DSA/RSA Authentication with SHA-1
15		SSH (Version 2.0)	[RFC 4253] (SSH v2.0)	n/a	no	With password authentication and guess success probability $\varepsilon \leq 10^{-8}$
16	RNG	Deterministic RNG DRG.2 (ssh-dflt)	OpenSSL default DRNG using a SHA-1 state transition and output function with 1024 bit internal state size.	n/a	n/a	
17		Deterministic RNG DRG.2 (ssh-fips)	ANSI X9.31 Appendix A2.4, AES 128 core	n/a	n/a	
18		Deterministic RNG DRG.2 (dm-init)	Linux random number generator accessible via the /dev/urandom device.	n/a	n/a	
19		Deterministic RNG DRG.2 (dm-run)	Linux random number generator accessible via the /dev/urandom device.	n/a	n/a	
20		Deterministic RNG DRG.2 (dm-fips)	ANSI X9.31 Appendix A2.4, AES 128 core	n/a	n/a	

Table 3: TOE cryptographic functionality

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 OSPs 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 product 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 of 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.

The limited validity for the usage of cryptographic algorithms as outlined in chapter 9 has to be considered by the user and his system risk management process.

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

Acronyms

ACL	Access Control List
AIS	Application Notes and Interpretations of the Scheme
API	Application Programming Interface
ASCII	American Standard Code for Information Interchange
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
cPP	Collaborative Protection Profile
DSA	Digital Signature Algorithm
EAL	Evaluation Assurance Level
ETR	Evaluation Technical Report
HLD	High Level Design
IPC	Interprocess Communication
IT	Information Technology
ITSEF	Information Technology Security Evaluation Facility
LAF	Lightweight Audit Framework
PAM	Pluggable Authentication Module
PP	Protection Profile
RELRO	RELocation Read-Only
RPM	RPM Package Manager
RSA	RSA (cryptosystem)
SAR	Security Assurance Requirement
SFP	Security Function Policy
SFR	Security Functional Requirement
SSHv2	Secure Shell Version 2
ST	Security Target
TCP/IP	Transmission Control Protocol / Internet Protocol
TLS	Transport Layer Security
TOE	Target of Evaluation

TSF TOE Security Functionality

Glossary

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

Collaborative Protection Profile - A Protection Profile collaboratively developed by an International Technical Community endorsed by the Management Committee.

dm_crypt - dm-crypt is a transparent disk encryption subsystem in Linux kernel.

Extension - The addition to an ST or PP of functional requirements not contained in CC part 2 and/or assurance requirements not contained in CC part 3.

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

Informal - Expressed in natural language.

Kickstart - Kickstart installations provide an automated alternative to the normal interactive installation.

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

OpenSSL - The Open Source toolkit for SSL/TLS

Package - named set of either security functional or security assurance requirements

Protection Profile - A formal document defined in CC, expressing an implementation independent set of security requirements for a category of IT Products that meet specific consumer needs.

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

Semiformal - Expressed in a restricted syntax language with defined semantics.

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

Target of Evaluation - An IT Product and its associated administrator and user guidance documentation that is the subject of an Evaluation.

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 4, September 2012
Part 2: Security functional components, Revision 4, September 2012
Part 3: Security assurance components, Revision 4, September 2012

[2] Common Methodology for Information Technology Security Evaluation (CEM), Evaluation Methodology, Version 3.1, Rev. 4, September 2012

[3] BSI certification: Technical information on the IT security certification of products, protection profiles and sites (BSI 7138) and Requirements regarding the Evaluation Facility for the Evaluation of Products, Protection Profiles and Sites under the CC and ITSEC (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-0924-2014, Version 2.6, 2014-08-12, Red Hat Enterprise Linux, Version 6.2 on 32 bit x86 Architecture, Red Hat, Inc. and atsec information security corp.

[7] Operating System Protection Profile, Version 2.0, 01 June 2010, BSI-CC-PP-0067-2010
OSPP Extended Package – Advanced Audit, Version 2.0, 28 May 2010

[8] Final Evaluation Technical Report, Version 4.0, 2014-11-14, BSI-DSZ-CC-0924_ETR_141114_v4.0, atsec information security GmbH, (confidential document)

[9] Configuration list for the TOE: CI list for source (rhel-62-logs.tgz), CI listing - Brew output of Git repositories of TOE packages (RHEL6.2-32bitcc-ci.tar.gz), CI list for the kernel (cc-rhel-6.2-kernel-2.6.32-220.45.1.el6.i686.rpm.txt) (confidential documents)

[10] Guidance documentation for the TOE, Version 2.10, 2014-10-10, EAL4 Evaluated Configuration Guide for Red Hat Enterprise Linux

[8] specifically

- AIS 20, Version 3, Funktionalitätsklassen und Evaluationsmethodologie für deterministische Zufallszahlengeneratoren
- AIS 31, Version 3, Funktionalitätsklassen und Evaluationsmethodologie für physikalische Zufallszahlengeneratoren
- AIS 32, Version 7, CC-Interpretationen im deutschen Zertifizierungsschema

This page is intentionally left blank.

C Excerpts from the Criteria

CC Part 1:

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
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

Assurance Class	Assurance Components
	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.

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."

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”

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.”

D Annexes

List of annexes of this certification report

Annex A: Security Target provided within a separate document.

This page is intentionally left blank.