National Information Assurance Partnership

Common Criteria Evaluation and Validation Scheme Validation Report

Cray UNICOS/mp Operating System
Version 2.4.15

Report Number: CCEVS-VR-04-0076
Dated: August 30, 2004
Version: 1.0

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

National Security Agency
Information Assurance Directorate
9800 Savage Road STE 6740
Fort George G. Meade, MD 20755-6740

ACKNOWLEDGEMENTS

Validation Team
Julie Evans, Paul Olson
National Security Agency
Ft. Meade, MD

Common Criteria Testing Laboratory
Evaluation Team
SAIC, Inc.
Columbia, MD

Table of Contents

1. Executive Summary

This report documents the NIAP validators' assessment of the CCEVS evaluation of the Cray UNICOS/mp operating system, Version 2.4.15. It presents the evaluation results, their justifications, and the conformance result.

The evaluation was performed by Science Applications International Corporation (SAIC) and was completed on August 11, 2004. The information in this report is largely derived from the Evaluation Technical Report (ETR) written by SAIC and submitted to the validators. The evaluation determined the product conforms to the CC Version 2.1, Part 2 and Part 3 to meet the requirements of Evaluation Assurance Level (EAL) 2 augmented with ALC_FLR.1. The target of evaluation (TOE) is claiming no Protection Profile conformance.

The Cray UNICOS/mp operating system under evaluation is an operating system that provides a scalable platform for heavy calculating applications not suited to more traditional computers. It supports multiple concurrent users, requiring them to be identified and authenticated, providing mechanisms that can be used to protect their data, and separating them from one another as well as from itself.

The Cray UNICOS/mp operating system is designed to operate on Cray proprietary X1 hardware that is included in the TOE.

The UNICOS/mp operating system provides a UNIX-based interface and command structure for the administration and use of the Cray X1.

The administrator guidance documents and product release notes provide the administrator with specific instructions to ensure that the product is delivered and installed in an appropriate environment. Note that while the information is available, Cray recommends and provides installation by Cray personnel.

The IT Environment comprises a data processing center suitable for a mainframe computer, and two sets of hardware attached to the TOE: the Cray X1 I/O architecture, and the Cray X1 support system.

The Cray X1 I/O architecture includes I/O drawers that communicate with various peripheral devices, Cray Programming Environment Server (CPES) that runs the tools used by the programmer for program development, Cray Network Subsystem (CNS) that connects a Cray X1 system to the customer's networks, and RAID subsystem that provides disk storage.

The Cray X1 support system consists primarily of the Cray Workstation that is used to administer the mainframe.

The validation team monitored the activities of the evaluation team, observed evaluation testing activities, provided guidance on technical issues and evaluation processes, reviewed selected evaluation evidence, and reviewed the individual work units and successive versions of the ETR.

The validation team found that the evaluation showed that the product satisfies all of the functional requirements and assurance requirements stated in the Security Target (ST). Therefore the validation team concludes that SAIC's findings are accurate, the conclusions justified, and the conformance results correct.

Disclaimers: The information contained in this Validation Report is not an endorsement of Cray by any agency of the U.S. Government and no warranty of Cray is either expressed or implied.

1.1 Evaluation Details

Item	Identifier
Evaluation Scheme	United States NIAP Common Criteria Evaluation and Validation Scheme
Dates of Evaluation	September 2003 – September 2004
Target of Evaluation	Cray UNICOS/mp Version 2.4.15
Protection Profile	None
Security Target	**Cray UNICOS/mp Operating System version 2.4.15 on Cray X1 hardware Security Target, Version 1.0, August 31, 2004**
Evaluation Technical Report	Evaluation Technical Report for the Cray UNICOS/mp Operating System, Version 2.4.15 on Cray X1 hardware, Version 1.0, August 31, 2004
Conformance Result	Part 2 conformant, Part 3 conformant, and EAL 2 augmented with ALC_FLR.1
Sponsor	Cray Inc.
Developer	Cray Inc.

1.2 Interpretations

It is specifically noted that no U.S. National Interpretations have been applied. The following International Interpretations have been identified where applied, but are not considered explicitly stated requirements:

- RI-3: Unique identification of configuration items in the configuration list
- RI-43: What does "clearly stated" mean?
- RI-51 (rev 1): Use of documentation without C & P elements
- RI-65: No component to call out security function management
- RI-84 : Aspects of objectives in TOE and environment
- RI-85 : SOF level is optional, not mandatory
- RI-103: Association Of Access Control Attributes With Subjects And Objects

2. Identification

2.1 TOE and TOE Identification

TOE: - Cray UNICOS/mp Version 2.4.15

CC Identification – *Common Criteria for Information Technology Security Evaluation*, Version 2.1, August 1999, ISO/IEC 15408.

CEM Identification – *Common Evaluation Methodology for Information Technology Security*, Part 1: Introduction and General Model, Version 0.6, January 1997; *Common Methodology for Information Technology Security Evaluation*, Part 2: Evaluation Methodology, Version 2.11, August 1999.

2.2 TOE Overview

The Target of Evaluation is a supercomputer, providing a scalable platform for heavy calculating applications not suited to more traditional computers and servers. Cray X1 systems utilize vector processors, shared memory, and a modernized vector instruction set in a highly scalable configuration. A Cray X1 system has high memory bandwidth and scalable system software.

The UNICOS/mp operating system provides a UNIX-based interface and command structure for the administration and use of the Cray X1.

3. TOE Security Objectives

3.1 Security Objectives for the TOE

The security objectives for the TOE are as follows:

- The TSF must ensure that only authorized users gain access to the TOE and its resources.
- The TSF must control access to resources based on identity of users. The TSF must allow authorized users to specify which users may access which resources.
- The TSF must be designed and implemented in a manner that ensures the organizational policies are enforced in the target environment.
- The TSF must provide all the functions and facilities necessary to support the authorized administrators who are responsible for the management of TOE security.

3.2 Clarification of Scope

The TOE security objectives were derived from the ST's organizational security policies and assumptions. The ST did not list any threats. Therefore, there are no threats that were not countered by the TOE.

The TOE does not comply with any published Protection Profile.

4. Assumptions

4.1 Usage Assumptions

The evaluation made the following assumptions concerning product usage:

- The processing resources of the TOE will be located within controlled access facilities, which will prevent unauthorized physical access.

- There will be one or more competent individuals assigned to manage the TOE and the security of the information it contains.

- The system administrative personnel are not careless, willfully negligent, or hostile, and will follow and abide by the instructions provided by the administrator documentation.

- Any other systems with which the TOE communicates are assumed to be under the same management control and operate under the same security policy constraints.

- The TOE hardware and software critical to security policy enforcement will be protected from unauthorized physical modification.

5. Security Policy

5.1 User data protection

The UNICOS/mp operating system enforces a discretionary access control (DAC) policy on all subjects and objects. Discretionary Access Control (DAC) is the mechanism by which an access to data is controlled based solely on the identity of the user and the resource. The implementation of DAC is accomplished by association of attributes, which are specific to the type of resource. This section first identifies the DAC attributes of subjects and each object. The DAC policies are specific to the type of object and are described following the attribute identification.

5.1.1 DAC Attributes

Following are the DAC-related attributes of a subject:

- Real UID
- Saved UID
- Saved GID
- Process group ID and session ID
- Process group leader ID
- Effective UID
- Real GID
- Effective GID
- Group list
- Process umask

Following are the DAC-related attributes of a file system object:

- Owner
- Owning Group
- Protection Mode

- Access Control List (optional)
- Default Access Control List (optional, relevant only for directories)

Following are the DAC-related attributes of a process object:

- Real User ID
- Effective User ID
- Saved User ID
- Process ID
- Process Group ID
- Process Group Structure
- Session Structure

5.1.2 DAC Permissions Checking for File System Objects

Permission checking relies on several object attributes. Two of those attributes are permission bits (i.e., protection mode) and Access Control Lists (ACLs). This section describes permission bits and ACLs and their access checking algorithms.

Permission bits divide accesses into three categories and users into three relative groups. The three categories of permissions are read, write, and execute. They are denoted as "r" for read, "w" for write, and "x" for execute. The three relative groups are the owner of the file, the owner's group, and every other user.

Permissions checking on permission bits checks for user permission first, followed by group, followed by other users. Permission is granted by the first permission set matched.

The UNICOS/mp operating system includes an additional DAC facility, Access Control Lists (ACLs) for files and directories. The ACL is an optional component of the DAC. ACLs are used to provide more fine-grained protection than the group permissions.

Any system file or directory may have an ACL that governs its discretionary access. This ACL is referred to as the access ACL for the file or directory. In addition, a directory may have an associated ACL that is applied to any file or subdirectory created within that directory. This ACL is referred to as a default ACL. The default ACL on a directory is not used to determine discretionary access to the directory. When a file or subdirectory is created within that directory, the file or subdirectory is assigned a new access ACL according to the following rules:

1) Starts with a copy of the parent directory's default ACL as its access ACL.

2) The user::, group::, other::, and mask:: entries are taken from the file's permission bits, with the process' umask applied.

3) If the parent directory's default ACL contains a mask entry, the access ACL mask entry is set from the file's group permission bits.

4) If the parent directory's default ACL does not contain a mask entry, then the new file's ACL group entry is set from the file's group permission bits.

5.1.3 DAC Permissions Checking for Process Objects

A process can always read and write its own data in user space. A process can also always read its own attributes. It can change its real UID and effective UID if the requested value is equal to its real or saved UID. A process may change its effective GID if the GID is equal to the real or saved GID.

There is one instance where processes access one another. One process can always read the attributes of another process if it can name the target process using its PID.

5.1.4 DAC Initialization and Revocation

For files and directories, if an ACL is used, permissions are assigned as described in Section 6.1.1.2 of the Security Target. If ACLs are not used, the creating process umask is used to assign initial permission. A umask contains the initial permission settings and may be set as restrictively as the subject wants. The various IDs associated with objects are taken from the creating subject's attributes. Process attributes are assigned at logon or are taken from a parent process. Object owners or administrators can modify the security attributes of objects.

If a subject changes the permissions of an object, the changes take affect on the next access check against the object. So, if a subject has a file descriptor open for an object and attempts to use the file descriptor, the old permissions will remain in affect until the subject closes the object. If the subject closes the object and then attempts to re-open it, the new permissions would then be enforced.

The user data protection function is designed to satisfy the following security functional requirements:

- FDP_ACC.1
- FDP_ACF.1
- FMT_SMR.1
- FMT_MSA.1

5.2 Identification and Authentication

When workstations are connected to the TOE, a user may request to authenticate (connecting through telnet) to the TOE to initiate processes on the TOE. This process prompts the user for an initial user name and then passes the name to the login command. The login command then prompts for a password. The login command will then continue to prompt for username / password pairs until a correct pair is entered or a configurable maximum number of attempts is exceeded. At this point, a configurable time delay occurs, and the login command exits. The user will not be allowed to perform any actions on the TOE until a successful authentication.

The /etc directory maintains the user attribute information. The identification and authentication subsystem maintains a file within the /etc directory that provides basic authentication data such as user name, user ID, and password. The user name may be an untrusted user or a role (i.e. non-administrative).

This information is supplemented by the following related database: group membership (/etc/group)

The user is never allowed to see the plain text form of a password or alter the password file directly. Only an administrator can manage user attributes (other than a user's password). When an administrator makes a change to a user attribute using the passmgmt command, the change takes affect on the next login. To ensure an immediate change, the administrator must require the user to logoff.

Once a user has been authenticated, process groups are used to associate users with subjects on the operating system. A subject is an entity within the TSC that causes operations to be performed. A process group is a set of processes that potentially all share the same address space. The degenerate case of a process group is a single process. A process is one thread of execution within a process group. Each process in a process group has its own stack. Every process in a process group potentially has access to the data (including the stack) of all the others. Other process attributes that may be shared include the address space, open file table, the current and root directories, the umask, the maximum file size, and the real and effective user identifiers (UIDs) and group identifiers (GIDs).

An existing subject can create a new subject by using the fork or exec system calls. A fork creates a new process that is a copy of the creating process. The new process is a child of the creator and has a new unique process ID. When a process within a process group performs an exec, it becomes the first process in a new process group and therefore, a new subject. A sproc system call is used to create a new process within an existing process group, but does not create a new subject unless and until the new process performs an exec system call, which changes its security attributes. All attributes are assigned to a subject at creation time.

The identification and authentication function is designed to satisfy the following security functional requirements:

- FIA_ATD.1
- FIA_UAU.2
- FIA_UID.2
- FIA_USB.1

5.3 Security management

The UNICOS/mp operating system supports the administrative role of root, the traditional UNIX administrative user. The root user is permitted to perform all administrative actions

including the ability to manage user authentication data and modify the default discretionary access policy. The terms UID=0, super user, root user and authorized administrator are used synonymously with root.

Users are allowed to change their own passwords using the passwd command. Passwords must be constructed to meet the following requirements:

- Each password must have at least six characters.
- Each password must contain at least two alphabetic characters and at least one numeric or special character. In this case, ``alphabetic'' means upper and lower case letters.
- Each password must differ from the user's login name and any reverse or circular shift of that login name. For comparison purposes, an upper case letter and its corresponding lower case letter are equivalent.
- New passwords must differ from the old by at least three characters. For comparison purposes, an upper case letter and its corresponding lower case letter are equivalent.

Administrators assign and can change any user's password using the same passwd command. User changes to passwords using the passwd command are constrained by password aging. Password aging means after a password change, the user may not change the password again for a site configured minimum period of time. This prevents the user from changing away from and back to an expired password too quickly. Once a password has been validated by the passwd command, it is encrypted and written into the password file. The clear text is never written into any file.

The security management function is designed to satisfy the following security functional requirements:

- FMT_MSA.1
- FMT_MSA.3
- FMT_MTD.1a
- FMT_MTD.1b
- FMT_MTD.1c
- FMT_SMF.1
- FMT_SMR.1

5.4 Protection of the TSF

The UNICOS/mp operating system executes instructions in two broad domains. Commands and applications execute in user-mode. In this mode, the memory management system and hardware restrictions on instruction execution isolate processes from each other, from operating system control data, and from direct hardware access. This establishes a strong

separation of the instruction streams and data contained in one process from those found in another process.

A thread within a share-group may initiate an operating system request from user-mode through the system call trap mechanism. A system call trap is a software interrupt operation that causes a context switch from user-mode into kernel-mode. In kernel-mode, the thread can only execute the kernel defined code sequence that follows from a system call. This kernel code sequence, however, has unrestricted direct access to kernel control data and the hardware. The kernel-mode and user-mode distinctions are enforced by the hardware.

A process in the UNICOS/mp operating system may have any of three levels of trust at any given time, depending on the code it executes. These three levels of trust define three trust domains for user-mode in the UNICOS/mp operating system:

- The user domain
- The administrative domain
- The system domain

The user domain contains processes without special authorization, operating on behalf of a single named user. Processes in the user domain may be simple or complex, but they cannot violate DAC restrictions, so they have no particular security function.

The administrative domain contains processes running with or without special authorization operating on behalf of administrators. Processes in the administrative domain always have the potential to assert administrative authority over policy restrictions. While these processes have a security function, that function is entirely under administrative control and, therefore, largely beyond comprehensive analysis. This limits analysis to correctness of function and potential administrative action.

The system domain contains processes executing with special authorization on behalf of potentially non-administrative users. This includes system daemons that provide services like authentication, batch processing, and so forth. It also includes applications that allow a user limited access to protected system resources, a password changing command, for example. Execution in the system domain implies policy enforcement by user-mode software. Policy enforcement requires well-defined and repeatable behavior, so analysis of system domain software includes a description of the controls placed by the software. Process isolation and memory protection features ensure that these processes have a protected execution environment.

The protection of the TSF function is designed to satisfy the following security functional requirements:

- FPT_RVM.1
- FPT_SEP.1

6. Architectural Information

The Cray X1 mainframe architecture includes the following components, which are housed in multiple cabinets.

- two or more interconnected nodes, where a node is a group of 4 multistreaming processors and shared memory, along with 4 high-bandwidth interconnection modules (SPC I/O Ports).

- RAID subsystem that provides disk storage for the Cray X1 system.

Multiple nodes can be interconnected with a minimum of 2 nodes required to create a useful Target of Evaluation (TOE).

The UNICOS/mp operating system that runs on the mainframe is based upon IRIX 6.5, modified and optimized for use on Cray X1 hardware. The UNICOS/mp operating system provides for separate hardware modes of operation, having kernel and user modes.

7. Documentation

7.1 Documents

Security Target - Cray UNICOS/mp Operating System version 2.4.15 on Cray X1 hardware Security Target, Version 1.0, August 31, 2004

Evaluation Technical Report - Evaluation Technical Report for the Cray UNICOS/mp Operating System, Version 2.4.15 on Cray X1 hardware, Version 1.0, August 31, 2004.

7.2 Results of the Evaluation

The Evaluation Team conducted the evaluation in accordance with the EAL 2 section of the CC and the CEM.

The Evaluation Team assigned a Pass, Fail, or Inconclusive verdict to each work unit of the EAL 2 assurance component. For Fail or Inconclusive work unit verdicts, the Evaluation Team advised the developer of the issue that needed to be resolved or the clarification that needed to be made to the particular evaluation evidence.

The Evaluation Team accomplished this by providing notes in the draft ETR sections for an evaluation activity (e.g., ASE) that recorded the Evaluation Team's evaluation results that the Team provided to the developer. The Evaluation Team also communicated with the developer by telephone, electronic mail, and meetings. If applicable, the Evaluation Team re-performed the work unit or units affected. In this way, the Evaluation Team assigned an overall Pass verdict to the assurance component only when all of the work units for that component had been assigned a Pass verdict. No constraints or assumptions were identified in performing this evaluation.

8. IT Product Testing

8.1 Developer Testing

The developer maintains a suite of tests to demonstrate that the product satisfies the claims that were made in the Security Target. The test documentation includes a test plan that describes the overall testing approach, and provides a mapping of the test procedures to the security functions.

The developer's overall testing approach can be summarized as follows:

- Where possible, the tests are designed to run automatically, without user intervention.
- Tests are self-contained and create/remove any resources required for validation (e.g., user ids, group ids, files).
- Test case source code contains descriptions of the test objective, the method of validation, and the expected result.
- Test cases are self-reporting and check the actual result against the expected result. Each test case reports a PASS/FAIL status and enough information to determine the objective of the test.

The developer provided the following documentation to substantiate their testing effort:

- Test plan.
- Test source.
- All open software problem reports documenting problems found by testing were attached to the test summary report.
- Test procedure information that documents how to set up and run all of the tests.
- Actual test result output.
- A test summary report that documents all testing activities.

The Security Functions that were tested are as follows:

1. **User data protection** that includes tests for:
   - Object owner
   - World Permission
   - Object Group Members
   - Matching ACL Entries: Testing the ACL Mask
   - Matching and Non-matching ACL Entries Tests
   - User Is Not A Member of the Effective Group
   - Combinations With Group ID and User ID Equal to 0 (zero)
   - ACLs Containing 256 Entries
2. **Identification and Authentication** – This functionality of the UNICOS/mp operating system is tested via manual tests for logging into the system. No automated tests are available for I & A.
3. **Security Management** - There are tests in the test suite for the DAC policy and for chmod and chown routines.
4. **Protection of the TSF** - The functionality of the system calls that complete protection of the TSF is verified by executing the UNICOS/mp regression test suite.

8.2 Evaluator Testing

The Evaluation Team conducted the evaluation in accordance with the EAL 2 section of the CC and the CEM. There are tests that correspond to each of the security functions. The evaluation team examined the coverage provided by Cray in order to identify additional potential evaluation team tests. Based on the test descriptions provided by Cray, the evaluation team expanded the scope of testing by addressing the security functional requirements more fully and also by examining additional interfaces.

The evaluation team exercised all of the developer's automated test cases twice. Explanations of other than entirely successful results were provided each time and some test database problems were addressed between the test runs resulting in a much cleaner set of results. While some issues remained after the second test run, the issues were explained to the evaluation team's satisfaction as being not security relevant and as such the evaluation team accepted the results of the developer's security tests.

The evaluation team exercised all of the developer's manual test procedures. The manual test procedures were related to identification and authentication. The evaluation team followed the test procedures without difficulty and experienced the expected test results. The evaluation team developed a number of manual test procedures, scripts, and programs to provide more comprehensive coverage of the security functions and their corresponding interfaces. The evaluation team also examined the developer's vulnerability analysis of the TOE looking for other obvious means by which to exploit the TOE. The evaluation team found that the TOE is resistant to potential attacks.

9. Evaluated Configuration

The test configuration consists of a Cray X1 running UNICOS/mp 2.4.15 preinstalled by the developer. Testing was conducted primarily from a remote location via telnet sessions established on the OS. The remote physical hardware was tested via the evaluation team's own equipment set up in the developer's software development facility.

Among the test procedures exercised, the evaluation team connected to the appropriate IP address identified during the physical inspection and then used UNICOS/mp interfaces to ensure that the MAC address and the proper product version were evident. This provided the evaluation team assurance that the correct system was being tested.

The evaluation team found that the TOE was set up in accordance with the 'evaluated configuration as described in the Security Target and the tests were installed according to the test documentation.

10. Validation Comments/Recommendations

The validation team recommends the Cray UNICOS/mp Version 2.4.15 receive an EAL 2 Certificate.

Users of the Cray X1 with UNICOS/mp interact with the system via a standard UNIX-style interface. Hence all available features are in the TOE. However, owners of the Cray X1 with UNICOS/mp should be aware that the TOE is a subset of the total product. As described in the Executive Summary above, there are some additional Cray products that will need to be set up and administered. The IT Environment comprises two sets of hardware: the Cray X1 I/O architecture, and the Cray X1 support system.

The Cray X1 I/O architecture includes the following components, which are housed in multiple cabinets, though only the RAID subsystem is included within the TOE:

- I/O drawers that convert the SPC protocol used by the TOE to Fibre Channel protocol used by various peripheral devices. Each drawer supports four SPC channels to the mainframe and up to 16 Fibre Channels to peripheral devices.
- Cray Programming Environment Server (CPES) that runs the tools used by the programmer for program development. The CPES provides the ability to offload compilation of code from the TOE, allowing for a distribution of programming environment processes away from the resources of the TOE. The CPES is based on a dedicated Sun Solaris server.
- Cray Network Subsystem (CNS) that connects a TOE to the customer's networks; supported protocols include Gigabit Ethernet and HIPPI. The CNS provides network connectivity for the TOE, packaging network packets into higher bandwidth chunks that are more efficient for the Cray X1 to handle, and reducing latency issues for the mainframe. The CNS is built on a Dell server running Linux.
- RAID subsystem that provides disk storage. This portion of the Cray X1 I/O Architecture is part of the TOE and was evaluated as such.

The Cray X1 support system consists primarily of the Cray Workstation (CWS), which is used to administer the mainframe. The CWS is built on a Sun Solaris server. The CPES and CNS are physically housed in one rack, possibly along with any attached RAID arrays.

11. Security Target

The Cray UNICOS/mp Operating System version 2.4.15 on Cray X1 hardware Security Target, Version 1.0, August 31, 2004 is included here by reference.

12. Abbreviations

Acronym	Phrase
ACL	Access Control List
CC	Common Criteria
CCEVS	Common Criteria Evaluation and Validation Scheme
CCTL	Common evaluation Testing Laboratory
CEM	Common Evaluation Methodology
CNS	Cray Network Subsystem
CPES	Cray Programming Environment Server
CWS	Cray Workstation
DAC	Discretionary Access Control
EAL	Evaluation Assurance Level
ETR	Evaluation Technical Report
GID	Group Identifier
HIPPI	ANSI High-Performance Parallel Interface (see RFC 2067)
I/O	Input/Output
IRIX 6.5	Predecessor operating system on which Cray –UNICOS/mp is based.
NIAP	National Information Assurance Program
NIST	National Institute of Standards & Technology
NSA	National Security Agency
NVLAP	National Voluntary Laboratory Accreditation Program
OR	Observation Report
PID	Process Identifier
PP	Protection Profile
RAID	Redundant Array of Independent Disks
SOF	Strength of Function
SPC	Simple Path Control Protocol
ST	Security Target
TOE	Target of Evaluation
TSC	TSF Scope of Control
TSF	TOE Security Functions
UID	User Identifier

13. Bibliography

The evaluation and validation methodology was drawn from the following:

[CC_PART1] Common Criteria for Information Technology Security Evaluation- Part 1: Introduction and general model, dated August 1999, Version 2.1.

[CC_PART2] Common Criteria for Information Technology Security Evaluation Part 2: Security functional requirements, dated August 1999, Version 2.1.

[CC_PART2A] Common Criteria for Information Technology Security Evaluation Part 2: Annexes, dated August 1999, Version 2.1.

[CC_PART3] Common Criteria for Information Technology Security Evaluation Part 3: Security assurance requirements, dated August 1999, Version 2.1.

[CEM_PART 1] Common Evaluation Methodology for Information Technology Security – Part 1: Introduction and general model, dated 1 November 1997, Version 0.6.

[CEM_PART2] Common Evaluation Methodology for Information Technology Security – Part 2: Evaluation Methodology, dated August 1999, Version 2.11.

[CCEVS_PUB1] Common Criteria, Evaluation and Validation Scheme for Information Technology Security, Organization, Management and Concept of Operations, Scheme Publication #1, Version 2.0 May 1999.

[CCEVS_PUB2] Common Criteria, Evaluation and Validation Scheme for Information Technology Security, Validation Body Standard Operating Procedures, Scheme Publication #2, Version 1.5, May 2000.

[CCEVS_PUB3] Common Criteria, Evaluation and Validation Scheme for Information Technology Security, Technical Oversight and Validation Procedures, Scheme Publication #3, Version 0.5, February 2001.

[CCEVS_PUB 4] Common Criteria, Evaluation and Validation Scheme for Information Technology Security, Guidance to Common Criteria Testing Laboratories, Scheme Publication #4, Version 1, March 20, 2001.

[CCEVS_PUB 5] Common Criteria, Evaluation and Validation Scheme for Information Technology Security, Guidance to Sponsors of IT Security Evaluations, Scheme Publication #5, Version 2.11, August 2000.