Technical Standard

COE Security Software Requirements Specification (SSRS)

The Open Group

Technical Standard

COE Security Software Requirements Specification (SSRS)

ISBN: 1-931624-31-3
Document Number: C035

Published in the U.K. by The Open Group, May 2003.

Any comments relating to the material contained in this document may be submitted to:

The Open Group
Apex Plaza
Forbury Road
Reading
Berkshire, RG1 1AX
United Kingdom

or by Electronic Mail to:

OGSpecs@opengroup.org

Contents

Preface

The Open Group

The Open Group, a vendor and technology-neutral consortium, has a vision of Boundaryless Information Flow achieved through global interoperability in a secure, reliable, and timely manner. The Open Group's mission is to drive the creation of Boundaryless Information Flow by:

- Working with customers to capture, understand, and address current and emerging requirements, establish policies, and share best practices
- Working with suppliers, consortia, and standards bodies to develop consensus and facilitate interoperability, to evolve and integrate open specifications and open source technologies
- Offering a comprehensive set of services to enhance the operational efficiency of consortia
- Developing and operating the industry's premier certification service and encouraging procurement of certified products

In the global eCommerce world of today, no single economic entity can achieve independence while still ensuring interoperability. The assurance that products will interoperate with each other across differing systems and platforms is essential to the success of eCommerce and business workflow. The Open Group, with its proven certification programs, is the international guarantor of interoperability in the new century.

The Open Group provides opportunities to exchange information and shape the future of IT. The Open Group members include some of the largest and most influential organizations in the world. The flexible structure of The Open Group membership allows for almost any organization, no matter what their size, to join and have a voice in shaping the future of the IT world.

More information is available at *www.opengroup.org*.

The Open Group has over 15 years' experience in developing and operating certification programs and has extensive experience developing and facilitating industry adoption of test suites used to validate conformance to an open standard or specification.

More information is available at *www.opengroup.org/testing*.

The Open Group publishes a wide range of technical documentation, the main part of which is focused on development of Technical and Product Standards and Guides, but which also includes white papers, technical studies, branding and testing documentation, and business titles. Full details and a catalog are available at *www.opengroup.org/pubs*.

As with all *live* documents, Technical Standards and Specifications require revision to align with new developments and associated international standards. To distinguish between revised specifications which are fully backwards-compatible and those which are not:

- A new *Version* indicates there is no change to the definitive information contained in the previous publication of that title, but additions/extensions are included. As such, it *replaces* the previous publication.
- A new *Issue* indicates there is substantive change to the definitive information contained in the previous publication of that title, and there may also be additions/extensions. As such, both previous and new documents are maintained as current publications.

Readers should note that Corrigenda may apply to any publication. Corrigenda information is published at *www.opengroup.org/corrigenda*.

This Document

This document was developed by the COE Forum and is based on the Defense Information Systems Agency (DISA), Common Operating Environment (COE) Platform Compliance Criteria, Security Software Requirements Specification (SSRS). It documents the security-related criteria for COE Platform Compliance.

The requirements in this document are grouped into the following categories:

1. Identification and Authentication (I&A)
2. Security Audit
3. Service Availability
4. Discretionary Access Control
5. Markings
6. Object Reuse
7. Data Confidentiality
8. System Integrity
9. System Architecture
10. Trusted Facility Management
11. Other Requirements

Trademarks

Boundaryless Information Flow is a trademark and UNIX and The Open Group are registered trademarks of The Open Group in the United States and other countries. All other trademarks are the property of their respective owners.

Acknowledgements

The Open Group gratefully acknowledges the Defense Information Systems Agency (DISA) as the original source of this material.

Referenced Documents

Normative references for this document are listed in Section 1.3 (on page 1).

Chapter 1

Introduction

1.1 Scope

This document identifies security-related criteria for COE Platform implementations.

These criteria are drawn from the Defense Systems Information Agency, Common Operating Environment (COE) Security Software Requirements Specification (SSRS).

The numbering of security requirements from the DISA SSRS is retained in this document as an aid to traceability.

1.2 Conformance

COE Platform implementations shall meet the criteria listed in this document. In some cases, text applies to system elements beyond the COE Platform implementation. In these cases, additional interpretation of the text is required to clarify the COE Platform implementation-related aspect of the text. Where interpretation is provided, Rationale text is provided below the requirement identified using *Italics*.

There are no requirements for the following sections of the DISA SSRS specification:

3.2.2	Trusted Path
3.2.6	Mandatory Access Control (MAC)
3.2.7	Sensitivity Labels
3.2.9	Trusted Interfaces
3.2.10	Object Reuse
3.2.14	Non-repudiation

1.3 Normative References

Defense Information Infrastructure (DII) Common Operating Environment (COE) Security Software Requirements Specification (SRS), Version 4.1, 15 October 1999.

1.4 Terminology

For the purposes of this document, the following terminology definitions apply:

can

Describes a permissible optional feature or behavior available to the user or application. The feature or behavior is mandatory for an implementation that conforms to this document. An application can rely on the existence of the feature or behavior.

implementation-defined

Describes a value or behavior that is not defined by this document but is selected by an implementor. The value or behavior may vary among implementations that conform to this document. An application should not rely on the existence of the value or behavior. An application that relies on such a value or behavior cannot be assured to be portable across conforming implementations.

The implementor shall document such a value or behavior so that it can be used correctly by an application.

legacy

Describes a feature or behavior that is being retained for compatibility with older applications, but which has limitations which make it inappropriate for developing portable applications. New applications should use alternative means of obtaining equivalent functionality.

may

Describes a feature or behavior that is optional for an implementation that conforms to this document. An application should not rely on the existence of the feature or behavior. An application that relies on such a feature or behavior cannot be assured to be portable across conforming implementations.

To avoid ambiguity, the opposite of *may* is expressed as *need not*, instead of *may not*.

shall

For an implementation that conforms to this document, describes a feature or behavior that is mandatory. An application can rely on the existence of the feature or behavior.

For an application or user, describes a behavior that is mandatory.

should

For an implementation that conforms to this document, describes a feature or behavior that is recommended but not mandatory. An application should not rely on the existence of the feature or behavior. An application that relies on such a feature or behavior cannot be assured to be portable across conforming implementations.

For an application, describes a feature or behavior that is recommended programming practice for optimum portability.

undefined

Describes the nature of a value or behavior not defined by this document which results from use of an invalid program construct or invalid data input.

The value or behavior may vary among implementations that conform to this document. An application should not rely on the existence or validity of the value or behavior. An application that relies on any particular value or behavior cannot be assured to be portable across conforming implementations.

unspecified

Describes the nature of a value or behavior not specified by this document which results from use of a valid program construct or valid data input.

The value or behavior may vary among implementations that conform to this document. An application should not rely on the existence or validity of the value or behavior. An application that relies on any particular value or behavior cannot be assured to be portable across conforming implementations.

1.5 Definitions

For the purposes of this document, the following definitions apply:

Login
The unspecified activity by which a user gains access to the system. Each login is associated with exactly one user name.

Supplier
A product vendor who is interested in, is applying for certification in, or has certified a product in The Open Group COE Platform Certification Program.

Trusted User
A user with appropriate privileges to administer the system.

User ID
A non-negative integer that is used to identify a system user.

Chapter 2

Identification and Authentication

3.2.1.1 The COE Platform implementation shall enforce individual accountability by providing the capability to uniquely identify each user to the system.

3.2.1.1.1 The COE Platform implementation shall require users to uniquely identify themselves before beginning to perform any actions that the system is expected to mediate.

This criteria is satisfied by the implementation if the requirement is met prior to loading Government-supplied software.

3.2.1.1.2 The COE Platform implementation shall require users to login prior to assuming a trusted profile (for example, system administrator, security officer, root user, and superuser).

3.2.1.2 Each user shall be uniquely identifiable (for example, user name or user ID) within an administrative domain.

This criteria is satisfied by the implementation if the requirement is met prior to loading Government-supplied software.

3.2.1.2.1 The COE Platform implementation shall uniquely identify each user for an entire enterprise.

This criteria is satisfied by the implementation if the requirement is met prior to loading Government-supplied software.

3.2.1.3 The COE Platform implementation shall provide the capability of associating the user's identity with all auditable actions taken by that individual.

3.2.1.4 The COE Platform implementation shall provide the following mechanism(s) to authenticate each user's identity:

3.2.1.4.1 The COE Platform implementation shall provide the capability to authenticate each user's identity with a password. Passwords shall meet the following requirements:

3.2.1.4.1.1 3.2.1.4.1.1.1 The COE Platform implementation shall provide a graphical user interface (GUI) for changing passwords.

3.2.1.4.1.1.2 The COE Platform implementation shall require a password be changed after the age of a password has exceeded a maximum of *n* days where *n* is configurable by a trusted user.

3.2.1.4.1.1.2.1 The default maximum days shall be 91.

3.2.1.4.1.1.3 The COE Platform implementation shall provide the capability to notify the user *n* days prior to password expiration where *n* is defined by a trusted user.

3.2.1.4.1.1.3.1 The COE Platform implementation shall default to notifying the user seven (7) days prior to password expiration.

3.2.1.4.1.1.4 The COE Platform implementation shall prohibit a password from being changed until the age of a password has exceeded a minimum of *n* days where *n* is defined by a trusted user.

3.2.1.4.1.1.4.1 The default minimum before a password can be changed shall be seven (7) days.

3.2.1.4.1.2 The COE Platform implementation shall permit a trusted user to override minimum password age limits when changing passwords.

3.2.1.4.1.4 The COE Platform implementation shall permit only trusted users to change passwords other than their own.

3.2.1.4.1.5 The COE Platform implementation shall provide the capability to require users to change a password during the initial use of a password created by trusted users.

3.2.1.4.1.7 The COE Platform implementation shall ensure that passwords feature specific characteristics configurable by a trusted user. The following characteristics shall be included:

3.2.1.4.1.7.1 Minimum password length

3.2.1.4.1.7.1.1 The default minimum password length shall be set to eight (8) characters.

A waiver of the requirement will be granted for six (6) character passwords if requested. Note that a note regarding this waiver will appear on the certificate.

3.2.1.4.1.7.2 Password character set (for example, alphanumeric plus special American National Standard Code for Information Interchange [ASCII] characters).

3.2.1.4.1.7.3 Password includes at least one numeric, case change, or special character (for example, 0-9, &, %).

3.2.1.4.1.8 The COE Platform implementation shall provide the capability to prohibit the following passwords:

3.2.1.4.1.8.2 Use of a user name within a password.

A waiver of the requirement will be granted if requested. Note that a note regarding this waiver will appear on the certificate.

3.2.1.4.5 The COE Platform implementation shall provide the capability where upon success user login the following information is displayed: the date and time of the last successful login and the number of unsuccessful login attempts since the last successful login.

This requirement is satisfied if the supplier provides a utility to output this information. In many implementations, the historical last command will satisfy the requirement. The capability must be present, but need not be implemented in the GUI login process.

3.2.1.4.5.1 The COE Platform implementation shall provide a trusted user with the capability to enable or disable display of the last successful login date and time and the number of unsuccessful login attempts.

3.2.1.5 The COE Platform implementation shall prevent unauthorized access to authentication data.

3.2.1.5.1 The COE Platform implementation shall prevent unauthorized disclosure of passwords during transmission across a network.

The GOTS APM software uses the Diffie-Hellman algorithm for encrypting network traffic within the administrative domain.

3.2.1.5.2 The COE Platform implementation shall prevent unauthorized disclosure of passwords while stored.

3.2.1.6 The COE Platform implementation shall provide the capability to limit invalid login attempts which are indicative of potential login attacks.

3.2.1.6.1 If the number of consecutive invalid login attempts for a single user ID reaches a threshold *n*, where *n* is configurable by a trusted user, the user ID shall be locked and will remain locked during all further login attempts with that user ID from within the administrative domain.

3.2.1.6.2 The COE Platform implementation shall be configurable by a trusted user to provide the capability to set the default number of consecutive login failures.

3.2.1.6.2.1 The default number of consecutive login failures shall be three (3).

3.2.1.6.3 The COE Platform implementation shall provide the capability for a trusted user, and only a trusted user, to disable the consecutive login failure functionality.

3.2.1.6.4 When a user ID is locked, the COE Platform implementation shall provide the capability to send a notification to a trusted user.

3.2.1.6.5 The COE Platform implementation shall provide the capability for a trusted user to restore locked user IDs.

This criteria is satisfied by the implementation if the requirement is met prior to loading Government-supplied software.

3.2.1.6.6 The COE Platform implementation shall perform login failure lockout for all login points (for example, console, remote login) in the administrative domain.

3.2.1.6.6.1 The COE Platform implementation shall perform login failure lockout for all login points (for example, console, remote login) in the enterprise.

Chapter 3

Security Audit

3.2.3.1 The COE Platform implementation shall provide the capability to create, maintain, process, and protect from modification or unauthorized access or destruction an audit trail of accesses to the objects it protects.

3.2.3.1.1 The COE Platform implementation shall protect audit data so that access to it is limited to those who are authorized to view audit data.

3.2.3.1.2 The COE Platform implementation shall protect the audit processes and audit data from change or deletion by general users. At a minimum, the COE Platform implementation shall protect the following:

3.2.3.1.2.1 Audit mechanisms (for example, executable files).

3.2.3.1.2.2 Configuration parameters (for example, audit configuration files).

3.2.3.1.2.3 Capability to enable or disable audit processes.

3.2.3.1.3 The COE Platform implementation shall provide a mechanism that generates a notification when the audit data has reached a configurable threshold of *n* percent of available storage capacity.

3.2.3.1.3.1 The COE Platform implementation shall be configurable by a trusted user to provide a capability for recovery in the event that the threshold *n* percent of available storage capacity has been exceeded. At a minimum, the following capabilities shall be provided:

3.2.3.1.3.1.2 Overwrite the oldest audit data.

3.2.3.1.3.1.4 Increase storage capacity for audit data.

Minimal compliance is satisfied by the ability to increase capacity manually via the Log File Manager.

3.2.3.1.3.2 The COE Platform implementation shall provide an interface for configuring which trusted user shall receive notifications when the audit data has reached the threshold *n* percent of available storage capacity.

3.2.3.1.3.3 The COE Platform implementation shall provide the capability for a trusted user to configure the threshold *n* percent of available storage capacity when a notification will be generated.

3.2.3.1.3.3.1 The default threshold *n* shall be 85 percent.

3.2.3.1.4 The COE Platform implementation shall provide a mechanism that generates a notification to a trusted user when the audit process(es) has failed.

3.2.3.1.4.2 The COE Platform implementation shall provide an interface for configuring which trusted user shall receive notifications when the audit process(es) has failed.

3.2.3.1.5 The COE Platform implementation shall provide a capability to archive and selectively retrieve audit data.

Minimal compliance is satisfied using commands (that is, tar, dd, and so on) at a command line. Neither a GUI nor automation is required.

3.2.3.1.5.1 The COE Platform implementation shall provide the capability to automatically archive audit data when the audit data reaches a configurable threshold of *n* percent of available storage capacity.

Minimal compliance is satisfied using commands (that is, tar, dd, and so on) at a command line. Neither a GUI nor automation (via Cron) is required.

3.2.3.1.5.4 The COE Platform implementation shall provide a mechanism that generates a time configurable notification to remind a trusted user (for example, a system administrator) to perform audit archive.

3.2.3.1.5.4.1 The COE Platform implementation shall provide a GUI for a trusted user to configure the time, represented as every *n* hours.

3.2.3.1.5.4.2 The default threshold *n* shall be every 168 hours.

3.2.3.2 The COE Platform implementation shall provide the capability to enable and disable auditable events.

3.2.3.3 The COE Platform implementation shall provide the capability to audit the following types of events:

3.2.3.3.1 Use of identification and authentication mechanisms.

3.2.3.3.2 Introduction of designated objects into a user's address space (for example, file open, program initiation).

3.2.3.3.3 Creation, modification, and deletion of designated objects.

3.2.3.3.4 Actions taken by trusted users.

3.2.3.3.7 Change in access control permissions.

3.2.3.3.9 System startup.

3.2.3.3.10 System shutdown.

3.2.3.4 The COE Platform implementation shall provide the capability for a trusted user to define security-relevant events.

3.2.3.5 For each recorded event, the COE Platform implementation shall identify in the audit record at least the following:

3.2.3.5.1 System date and time (to the nearest second) of the event.

3.2.3.5.2 User ID.

3.2.3.5.3 Type of event.

3.2.3.5.4 Success or failure of the event.

3.2.3.6 For identification and authentication events, the audit record shall identify the origin of the request (for example, terminal ID, host IP address).

3.2.3.10 The COE Platform implementation shall provide the capability to receive application-level audit data (for example, the UNIX *syslog* logging facility, Windows NT event log).

3.2.3.11 The COE Platform implementation shall provide the capability to generate reports of audit data that has been collected.

3.2.3.11.1 The COE Platform implementation shall provide the capability to generate reports based on fields in event records or Boolean combinations of those fields.

3.2.3.11.2 The COE Platform implementation shall provide the capability to generate reports based on ranges of system date and time that audit records were collected.

Chapter 4

Availability

3.2.4.1 The COE Platform implementation shall be capable of detecting the failure of a system service or resource.

Minimally satisfied by POST on boot.

3.2.4.1.2 The COE Platform implementation shall provide the following capabilities to notify a trusted user:

3.2.4.2 Upon recovery of a failed system resource, the COE Platform implementation shall verify that it returns in a secure state.

Minimally satisfied by POST on boot.

3.2.4.2.1 Upon recovery of a failed system resource, the COE Platform implementation shall provide the capability to determine whether file systems are intact.

Minimally satisfied by the fsck utility or equivalent.

3.2.4.2.2 Upon recovery of a failed system resource, the COE Platform implementation shall provide the capability to determine whether access control permissions are unchanged from the state prior to the failure.

Minimally satisfied by the Tripwire[1] *tool. The supplier may propose an equivalent for review.*

3.2.4.2.3 Upon recovery of a failed system resource, the COE Platform implementation shall ensure that user privileges have not increased.

Minimally satisfied by the Tripwire tool. The supplier may propose an equivalent for review.

3.2.4.3 The COE Platform implementation shall provide the capability for a trusted user to selectively revoke a user's access to services.

Minimally satisfied by the combination of TCPwrapper[2] *and DAC.*

3.2.4.3.1 The COE Platform implementation shall provide the capability to kill or halt a user's process(es).

3.2.4.4 The COE Platform implementation shall provide the capability to perform system and database backups.

System Backup/Restore capability required. The supplier must identify a solution for review.

3.2.4.4.1 The COE Platform implementation shall provide the capability to scan for viruses during backup operations.

[1] Tripwire is a tool that checks to see what has changed on your system. The program monitors key attributes of files that should not change, including binary signature, size, expected change of size, and so on. Refer to Tripwire, Inc. at *www.tripwire.com* and *www.tripwire.org* for the commercial and Open Source versions of the Tripwire tool.

[2] TCPwrapper is the common name for Wietse Venema's *tcpd*. It gives a system administrator the ability to block and/or log access attempts via *tcp*. This provides an additional level of protection inside a firewall and increases the granularity of security to the system level, without having to control the firewall. See *ftp://ftp.porcupine.org/pub/security/index.html*.

Virus Scan capability required. The supplier must identify a solution for review.

3.2.4.5 The COE Platform implementation shall provide the capability to recover from failures using system and database backups.

System Backup/Restore capability required. The supplier must identify a solution for review.

Chapter 5

Discretionary Access Control (DAC)

3.2.5.1 The COE Platform implementation shall provide the capability to define access between named users and/or defined sets of users and named objects (for example, files, database elements, and programs).

3.2.5.2 The COE Platform implementation shall provide the capability to control access between named users and/or defined sets of users and named objects (for example, files, database elements, and programs).

3.2.5.3 The COE Platform implementation shall restrict access to objects based on the user's and/or defined sets of user's identity and on access rights (for example, read, write, execute).

3.2.5.3.1 The COE Platform implementation shall provide the capability to restrict access to objects based on the user's role.

3.2.5.3.2 The COE Platform implementation shall provide the capability to restrict access to objects based on the user's organization.

3.2.5.4 The COE Platform implementation shall provide the capability for users to specify and control sharing of objects by named users or defined sets of users (for example, UNIX groups, access control lists), or by both.

3.2.5.5 The COE Platform implementation shall provide controls to limit the propagation of access rights.

3.2.5.6 The COE Platform implementation shall, either by explicit user action or by default, protect objects from unauthorized access.

3.2.5.7 The COE Platform implementation shall provide the capability to assign access rights to authorized users.

3.2.5.8 The COE Platform implementation shall permit a user to grant or revoke access to an object if the user has control permission (for example, file owner) for that object.

3.2.5.9 The COE Platform implementation shall provide a means to associate applications with a work environment (that is, profiles) and allow users to specify the work environment (that is, profile selection) during a session.

3.2.5.9.1 The COE Platform implementation shall permit a user to hold membership in multiple groups of users simultaneously and have all the access rights of those groups.

3.2.5.11 The COE Platform implementation shall be capable of restricting access to input/output (I/O) devices (for example, floppy disks and tape drives).

3.2.5.11.1 The COE Platform implementation shall provide a capability to specify which users may access which I/O devices.

3.2.5.12 The COE Platform implementation shall provide a *deadman* capability that is activated if user input devices have been idle for longer than a time period of *n* minutes, where *n* is configurable by a trusted user (for example, a system administrator).

3.2.5.12.1 When the *deadman* capability is activated after *n* minutes, the COE Platform implementation shall discontinue the user session (log the user off).

3.2.5.12.2 The configurable time period *n* shall default to 30 minutes.

3.2.5.16 The COE Platform implementation shall provide a screen-lock capability that is activated if user input devices have been idle for longer than a time period of *n* minutes, where *n* is configurable by a trusted user (for example, a system administrator).

3.2.5.16.1 When the screen-lock capability is activated after *n* minutes, the COE Platform implementation shall screen-lock the terminal and display a selected screensaver.

3.2.5.16.2 The configurable time period *n* shall default to 15 minutes.

3.2.5.16.5 Any user-input device shall be used to initiate actions to restore a screen-locked terminal.

3.2.5.16.6 The specific input value (whether from keyboard, mouse, or other input device) used to restore a screen-locked terminal shall be ignored except to initiate actions to unlock the terminal.

3.2.5.16.7 The COE Platform implementation shall require that users re-authenticate themselves to unlock a screen-locked terminal.

3.2.5.16.8 The screen-lock capability shall be available for users to activate via icon, menu selection, or button.

3.2.5.16.9 The COE Platform implementation shall provide the capability for a trusted user (for example, a system administrator) to unlock a screen-locked terminal irrespective of which user was logged in to that terminal.

Chapter 6

Markings

3.2.8.2 The COE Platform implementation shall display a security warning during the login process to indicate that misuse of the system is subject to applicable penalties.

3.2.8.2.1 This security warning shall state that the user accepts responsibility for his or her actions prior to being permitted to access information.

Chapter 7

Object Reuse

3.2.10.1 The COE Platform implementation shall ensure that no information, including encrypted representations of information, produced by a prior subject's actions is made available to any subject that obtains access to an object that has been released back to the COE Platform implementation.

Vendor may demonstrate or present an analysis supporting a claim of compliance. Applies to disk and memory.

3.2.10.2 The COE Platform implementation shall ensure that all authorizations to information contained within a storage object have been revoked prior to initial assignment, allocation, or reallocation to a subject from the COE Platform implementation's pool of unused storage objects.

Vendor may demonstrate or present an analysis supporting a claim of compliance. Applies to disk and memory.

Chapter 8

Data Confidentiality

3.2.11.1 The COE Platform implementation shall provide an interface to cryptographic application programming interfaces for use by applications to selectively encrypt and decrypt data and files.

Minimal compliance is provided by the ‘‘crypt’’ implementation.

Chapter 9

Data Integrity

3.2.12.1 The COE Platform implementation shall provide the capability to detect unauthorized modification or destruction of data during storage (for example, using digital signatures and hash codes on files).

Minimally satisfied by the Tripwire tool or equivalent.

3.2.12.1.1 The COE Platform implementation shall provide the capability to audit unauthorized modification or destruction of data during storage.

Minimally satisfied by the Tripwire tool or equivalent.

Chapter 10

System Integrity

3.2.13.1 The COE Platform implementation shall provide the capability to validate the correct operation of the hardware, software, and firmware elements of the system's security services.

Minimally satisfied by POST.

3.2.13.2 The COE Platform implementation shall provide the capability to automatically validate the correct operation of the hardware and firmware elements of the COE security services during recovery from failure.

Minimally satisfied by POST on restart.

3.2.13.3 The COE Platform implementation shall be configured such that a password must be entered to boot to a privileged start-up state.

3.2.13.4 The COE Platform implementation shall provide the capability to detect and eradicate malicious code (for example, viruses).

Virus Scan capability required. The supplier must identify a solution for review.

3.2.13.4.1 The COE Platform implementation shall provide the capability for a user to initiate a scan of hard drives and removable media for malicious code and alert the user and a trusted user if such code is detected.

Requirement should be interpreted to allow a trusted user only to initiate such a scan. (Normal user shall not access full file system). Virus Scan capability required. The supplier must identify a solution for review.

3.2.13.4.2 The COE Platform implementation shall provide the capability to automatically scan hard drives and removable media for malicious code.

Minimally satisfied by Cron invocation of Virus Scan capability. The supplier must identify a solution for review.

3.2.13.4.3 The COE Platform implementation shall provide the capability to alert the user and trusted user of the detection of malicious code by the following techniques:

3.2.13.4.3.1 Visible message on the workstation screen.

Capability required. The supplier must identify an equivalent solution for review.

3.2.13.4.3.2 Audible alarm.

Capability required. The supplier must identify an equivalent solution for review.

3.2.13.4.4 The COE Platform implementation shall provide the capability to create, maintain, and update a virus database to support virus detection and eradication.

Capability required. The supplier must identify an equivalent solution for review.

3.2.13.4.5 The COE Platform implementation shall provide the capability to capture malicious code (for example, virus) during the eradication process and store the malicious code as data in a separate file.

Virus Scan capability required. The supplier must identify a solution for review.

3.2.15.1 The COE Platform implementation security services shall protect themselves from external interference or tampering (for example, by modification of their code or data structures).

Minimal compliance via DAC.

3.2.15.2 The COE Platform implementation shall isolate resources to be protected so that they are subject to the access control requirements.

Minimal compliance via DAC.

3.2.15.3 The COE Platform implementation shall implement the principle of least privilege such that each subject is granted the most restrictive set of privileges needed for the performance of authorized tasks.

Minimal compliance via DAC.

Glossary

ACL
Access Control List

API
Application Program Interface

APM
Account and Profile Manager

CDE
Common Desktop Environment

CDS
Common Data Store

CITI
Center for Information Technology Integration

COE
Common Operating Environment

COTS
Commercial off-the-shelf

DISA
Defense Information Systems Agency

DNS
Domain Name Service

EEPROM
Electronically Erasable Programmable Read-Only Memory

FTP
File Transfer Protocol

GIG
Global Information Grid

GOTS
Government off-the-shelf

GSPR
Global Software Problem Report

GUI
Graphical User Interface

HTML
Hypertext Markup Language

HTTP
Hypertext Transfer Protocol

ID
Identification

IP
Installation Procedures (referring to documentation)

IP
Internet Protocol (as in IP address)

JPL
Jet Propulsion Laboratory

NFS
Network File System

NIS+
Network Information Service Plus

PAM
Pluggable Authentication Modules

PDC
Primary Domain Controller

PDF
Portable Document Format

PGRM
Programmer's Guide and Reference Manual

PID
Process ID

POSIX
Portable Operating System Interface for UNIX

PSM
PAM Strike Manager

SAM
System Administrator's Manual

SECAM
Security Administrator's Manual

SMTP
Simple Mail Transfer Protocol

STD
Software Test Description

STR
Software Test Report

SVD
Software Version Description

TCP/IP
Transmission Control Protocol / Internet Protocol