Common Criteria Evaluation and Validation Scheme

National Information Assurance Partnership (NIAP)

Title: Application Software Essential Security Requirements

Maintained by: National Information Assurance Partnership (NIAP)

Unique Identifier: 42

Version: 1.0

Status: draft

Date of issue: 29 Aug 2015

Approved by:

Supersedes:

Status

The following is an Essential Security Requirements (ESR) document for application software. The creation of an ESR is a necessary prerequisite to develop an Application Software cPP, and this document represents material provided by NIAP for that purpose.

Background and Purpose

This document describes a core set of security requirements for application software. These requirements cover basic security behavior for application software. Evaluation against the resulting Protection Profile ensures that this fundamental set of requirements is met. These fundamental requirements must be extended to adequately cover the functionality of many types of applications.

> **Note:** *This is not a declaration that all software should be evaluated through the Common Criteria. Although it depends on the national market, Common Criteria evaluation generally focuses on providing assurance for products which provide security functionality. Many applications without security functionality, particularly on mobile platforms, now receive some type of evaluation (often called vetting). This occurs because software without security functionality, when flawed, has security consequences. This document is offered as a reference for those activities, in addition to its role in the Common Criteria. The goal is to establish a consistent set of expectations for all application software developers, independent of the evaluation methodology.*

The vast majority of application software should satisfy this core set of requirements, yet a very small set of extremely specialized software may not do so. The requirements for such exceptional software may be specified in Protection Profiles which do not extend the requirements described here.

Application software in the context of this document is software that runs on a platform and performs tasks on behalf of the user or owner of the system. The platform for the application is an operating system, an execution environment, or some combination of these.

Use Cases

Application software is used in innumerable specific use cases.

However, in formal Common Criteria evaluations we seek evaluation only of applications which provide security functionality (which are called IA or IA-enabled in some markets). Such applications include thin clients and host-based security agents. Other applications will be covered by Extended Packages of the resulting Protection Profile (email client, web browser, VPN client, MDM agent).

Resources to be protected

- Sensitive data in transit.
- Sensitive data stored locally by the application.
- Application code and configuration parameters.

The application should also not require security features in the platform be disabled, as this weakens the underlying platform.

Attacker access

- An attacker is assumed to attempt attacks from the following vantage points:
  - The network across which the application engages in communication, both actively and passively.
  - The platform on which the application is installed, though as an unprivileged subject.
- An attacker has an arbitrary amount of time to analyze the behavior of the application, its interaction with its host device or platform, and/or the data it transmits over the network.

Evaluation Boundary

- The application consists of the software provided by its vendor. Any software in the application installation package is potentially in scope during evaluation. This includes those pieces that may extend the functionality of the underlying platform, such as kernel drivers. The application exists both as an object that is stored on the file system of the host platform as well as a runtime object that exists during its execution. The application code may execute directly on a microprocessor, or it may be script or bytecode interpreted by a runtime environment.
- Shared libraries (static or dynamically loaded) from third parties that convey with the application are also in scope.

Essential Security Requirements

Functionality-related requirements are:

- Limit network connectivity to necessary communications, and encrypt sensitive data that is transmitted remotely using a trusted communications channel.
- Leverage the platform to protect any sensitive data at rest stored in non-volatile memory, such as credentials.
- Require initial assignment of credentials by the end user whenever the application is shipped with default credentials or no credentials.
- Restrict access to those platform resources which are necessary to achieve its stated functionality.
- Properly implement, or leverage the platform, for cryptographic operations such as key generation, encryption and decryption, random bit generation, hashing, signing, and keyed-hash message authentication.
- Leverage the platform's exploit mitigation features, and never engage in behavior that undermines the platform's security features.
- Be distributed only in the format supported by the platform's package manager, and ensure trusted update.

Assurance-related requirements are:

- Timely patching of any publicly-disclosed vulnerabilities, including those in 3rd party components that convey with the application.
- Use of anti-exploitation options provided in the development toolchain.

Assumptions

- The application relies upon a trustworthy computing platform for its execution. This includes the underlying platform and whatever runtime environment it provides to the application.
- The user of the application software is not willfully negligent or hostile, and uses the software in compliance with the applied enterprise security policy.
- The administrator of the underlying platform or application software is not careless, willfully negligent or hostile, and administers the software within compliance of the approved enterprise security policy.

Optional Extensions

- Client authentication to remote peers using X.509v3 certificates.

Objective Requirements

- Use of Software ID (SWID) tags to enable software inventory as defined by ISO/IEC 19770-2:2009.

Outside the Scope of Evaluation

- The hardware or firmware of the underlying platform.
- The host operating system or runtime environment on which the application executes.
- Specific functional behavior that is not global to all applications.