CWE-653: Improper Isolation or Compartmentalization

Weakness ID: 653
Vulnerability Mapping: ALLOWED
Abstraction: Class

View customized information: Conceptual | Operational | Mapping Friendly | **Complete** | Custom

Description

The product does not properly compartmentalize or isolate functionality, processes, or resources that require different privilege levels, rights, or permissions.

Extended Description

When a weakness occurs in functionality that is accessible by lower-privileged users, then without strong boundaries, an attack might extend the scope of the damage to higher-privileged users.

Alternate Terms

Separation of Privilege: Some people and publications use the term "Separation of Privilege" to describe this weakness, but this term has dual meanings in current usage. This node conflicts with the original definition of "Separation of Privilege" by Saltzer and Schroeder; that original definition is more closely associated with CWE-654. Because there are multiple interpretations, use of the "Separation of Privilege" term is discouraged.

Common Consequences

Scope	Impact	Likelihood
Access Control	**Technical Impact:** *Gain Privileges or Assume Identity; Bypass Protection Mechanism* The exploitation of a weakness in low-privileged areas of the software can be leveraged to reach higher-privileged areas without having to overcome any additional obstacles.	

Potential Mitigations

Phase: Architecture and Design

Break up privileges between different modules, objects, or entities. Minimize the interfaces between modules and require strong access control between them.

Relationships

Relevant to the view "Research Concepts" (CWE-1000)

Nature	Type	ID	Name
ChildOf	P	693	Protection Mechanism Failure
ChildOf	C	657	Violation of Secure Design Principles
ParentOf	B	1189	Improper Isolation of Shared Resources on System-on-a-Chip (SoC)
ParentOf	B	1331	Improper Isolation of Shared Resources in Network On Chip (NoC)

Relevant to the view "Software Development" (CWE-699)

Nature	Type	ID	Name
MemberOf	C	1212	Authorization Errors

Relevant to the view "Architectural Concepts" (CWE-1008)

Modes Of Introduction

Phase	Note
Architecture and Design Implementation	COMMISSION: This weakness refers to an incorrect design related to an architectural security tactic.

Applicable Platforms

Languages

Class: Not Language-Specific *(Undetermined Prevalence)*

Demonstrative Examples

Example 1

Single sign-on technology is intended to make it easier for users to access multiple resources or domains without having to authenticate each time. While this is highly convenient for the user and attempts to address problems with psychological acceptability, it also means that a compromise of a user's credentials can provide immediate access to all other resources or domains.

Example 2

The traditional UNIX privilege model provides root with arbitrary access to all resources, but root is frequently the only user that has privileges. As a result, administrative tasks require root privileges, even if those tasks are limited to a small area, such as updating user manpages. Some UNIX flavors have a "bin" user that is the owner of system executables, but since root relies on executables owned by bin, a compromise of the bin account can be leveraged for root privileges by modifying a bin-owned executable, such as CVE-2007-4238.

Observed Examples

Reference	Description
CVE-2021-33096	Improper isolation of shared resource in a network-on-chip leads to denial of service
CVE-2019-6260	Baseboard Management Controller (BMC) device implements Advanced High-performance Bus (AHB) bridges that do not require authentication for arbitrary read and write access to the BMC's physical address space from the host, and possibly the network [REF-1138].

Weakness Ordinalities

Ordinality	Description
Primary	*(where the weakness exists independent of other weaknesses)*

Detection Methods

Automated Static Analysis - Binary or Bytecode

According to SOAR, the following detection techniques may be useful:

Cost effective for partial coverage:

- Compare binary / bytecode to application permission manifest

Effectiveness: SOAR Partial

Manual Static Analysis - Source Code

According to SOAR, the following detection techniques may be useful:

Highly cost effective:

- Manual Source Code Review (not inspections)

Cost effective for partial coverage:

- Focused Manual Spotcheck - Focused manual analysis of source

Effectiveness: High

Architecture or Design Review

According to SOAR, the following detection techniques may be useful:

Highly cost effective:

- Inspection (IEEE 1028 standard) (can apply to requirements, design, source code, etc.)
- Formal Methods / Correct-By-Construction

Cost effective for partial coverage:

- Attack Modeling

Effectiveness: High

Memberships

Nature	Type	ID	Name
MemberOf	C	901	SFP Primary Cluster: Privilege
MemberOf	C	1348	OWASP Top Ten 2021 Category A04:2021 - Insecure Design
MemberOf	C	1418	Comprehensive Categorization: Violation of Secure Design Principles

Vulnerability Mapping Notes

Usage: ALLOWED *(this CWE ID could be used to map to real-world vulnerabilities)*

Reason: Acceptable-Use

Rationale:

This CWE entry is at the Base level of abstraction, which is a preferred level of abstraction for mapping to the root causes of vulnerabilities.

Comments:

Carefully read both the name and description to ensure that this mapping is an appropriate fit. Do not try to 'force' a mapping to a lower-level Base/Variant simply to comply with this preferred level of abstraction.

Notes

Relationship

There is a close association with CWE-250 (Execution with Unnecessary Privileges). CWE-653 is about providing separate components for each "privilege"; CWE-250 is about ensuring that each component has the least amount of privileges possible. In this fashion, compartmentalization becomes one mechanism for reducing privileges.

Terminology

The term "Separation of Privilege" is used in several different ways in the industry, but they generally combine two closely related principles: compartmentalization (this node) and using only one factor in a security decision (CWE-654). Proper compartmentalization implicitly introduces multiple factors into a security decision, but there can be cases in which multiple factors are required for authentication or other mechanisms that do not involve compartmentalization, such as performing all required checks on a submitted certificate. It is likely that CWE-653 and CWE-654 will provoke further discussion.

References

[REF-196] Jerome H. Saltzer and Michael D. Schroeder. "The Protection of Information in Computer Systems". Proceedings of the IEEE 63. 1975-09. <http://web.mit.edu/Saltzer/www/publications/protection/>.

[REF-535] Sean Barnum and Michael Gegick. "Separation of Privilege". 2005-12-06. <https://web.archive.org/web/20220126060047/https://www.cisa.gov/uscert/bsi/articles/knowledge/principles/separation-of-privilege>. *URL validated: 2023-04-07.*

[REF-1138] Stewart Smith. "CVE-2019-6260: Gaining control of BMC from the host processor". 2019. <https://www.flamingspork.com/blog/2019/01/23/cve-2019-6260:-gaining-control-of-bmc-from-the-host-processor/>.

Content History

Submissions

Submission Date	Submitter	Organization
2008-01-18 *(CWE Draft 8, 2008-01-30)*	Pascal Meunier	Purdue University

Modifications

Previous Entry Names

Page Last Updated: July 16, 2024