**What Is SANS Top 25?**

The CWE/ SANS top 25 most dangerous software flaws are a list of the **most dangerous flaws** because they let attackers gain complete control of the software, steal data and information from it, or prohibit it from functioning at all.

The SANS top 25 is a versatile starting point that can be used by almost any organization, regardless of size, industry, geography, or**government/ commercial** status.

The controls are prioritized to protect the organization’s infrastructure and data by strengthening the organization’s defense system through continuous automated protection and monitoring. They were developed and maintained by an international group of **organizations, government agencies, and security experts**.

### List Of SANS Top 25:

**Rank:**1 **ID:** [CWE-787](https://cwe.mitre.org/data/definitions/787.html)

**Name:** Out-of-bounds Write

**Description:** The product writes data past the end, or before the beginning, of the intended buffer.

**Rank:** 2 **ID:** [CWE-79](https://cwe.mitre.org/data/definitions/79.html)

**Name:** Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')

**Description:** The product does not neutralize or incorrectly neutralize user-controllable input before it is placed in output that is used as a web page that is served to other users.

**Rank:** 3 **ID:** [CWE-89](https://cwe.mitre.org/data/definitions/89.html)

**Name:** Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')

**Description:** The product constructs all or part of an SQL command using externally influenced input from an upstream component, but it does not neutralize or incorrectly neutralize special elements that could modify the intended SQL command when it is sent to a downstream component.

**Rank:** 4 **ID:** [CWE-416](https://cwe.mitre.org/data/definitions/416.html)

**Name:** Use After Free

**Description:** Referencing memory after it has been freed can cause a program to crash, use unexpected values, or execute code.

**Rank:** 5 **ID:** [CWE-78](https://cwe.mitre.org/data/definitions/78.html)

**Name:** Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')

**Description:** The product constructs all or part of an OS command using externally influenced input from an upstream component, but it does not neutralize or incorrectly neutralize special elements that could modify the intended OS command when it is sent to a downstream component.

**Rank:** 6 **ID:** [CWE-20](https://cwe.mitre.org/data/definitions/20.html)

**Name:** Improper Input Validation

**Description:** The product receives input or data, but it does not validate or incorrectly validate that the input has the properties that are required to process the data safely and correctly.

**Rank:** 7 **ID:** [CWE-125](https://cwe.mitre.org/data/definitions/125.html)

**Name:** Out-of-bounds Read

**Description:** The product reads data past the end, or before the beginning, of the intended buffer.

**Rank:** 8 **ID:** [CWE-22](https://cwe.mitre.org/data/definitions/22.html)

**Name:** Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal')

**Description:** The product uses external input to construct a pathname that is intended to identify a file or directory that is located underneath a restricted parent directory, but the product does not properly neutralize special elements within the pathname that can cause the pathname to resolve to a location that is outside of the restricted directory.

**Rank:** 9 **ID:** [CWE-352](https://cwe.mitre.org/data/definitions/352.html)

**Name:** Cross-Site Request Forgery (CSRF)

**Description:** The web application does not, or can not, sufficiently verify whether a well-formed, valid, consistent request was intentionally provided by the user who submitted the request.

**Rank:** 10 **ID:** [CWE-434](https://cwe.mitre.org/data/definitions/434.html)

**Name:** Unrestricted Upload of File with Dangerous Type

**Description:** The product allows the attacker to upload or transfer files of dangerous types that can be automatically processed within the product's environment.

**Rank:** 11 **ID:** [CWE-862](https://cwe.mitre.org/data/definitions/862.html)

**Name:** Missing Authorization

**Description:** The product does not perform an authorization check when an actor attempts to access a resource or perform an action.

**Rank:** 12 **ID:** [CWE-476](https://cwe.mitre.org/data/definitions/476.html)

**Name:** NULL Pointer Dereference

**Description:** A NULL pointer dereference occurs when the application dereferences a pointer that it expects to be valid, but is NULL, typically causing a crash or exit.

**Rank:** 13 **ID:** [CWE-287](https://cwe.mitre.org/data/definitions/287.html)

**Name:** Improper Authentication

**Description:** When an actor claims to have a given identity, the product does not prove or insufficiently proves that the claim is correct.

**Rank:** 14 **ID:** [CWE-190](https://cwe.mitre.org/data/definitions/190.html)

**Name:** Integer Overflow or Wraparound

**Description:** The product performs a calculation that can produce an integer overflow or wraparound when the logic assumes that the resulting value will always be larger than the original value. This can introduce other weaknesses when the calculation is used for resource management or execution control.

**Rank:** 15 **ID:** [CWE-502](https://cwe.mitre.org/data/definitions/502.html)

**Name:** Deserialization of Untrusted Data

**Description:** The product deserializes untrusted data without sufficiently verifying that the resulting data will be valid.

**Rank:** 16 **ID:** [CWE-77](https://cwe.mitre.org/data/definitions/77.html)

**Name:** Improper Neutralization of Special Elements Used in a Command ('Command Injection')

**Description:** The product constructs all or part of a command using externally influenced input from an upstream component, but it does not neutralize or incorrectly neutralize special elements that could modify the intended command when it is sent to a downstream component.

**Rank:** 17 **ID:** [CWE-119](https://cwe.mitre.org/data/definitions/119.html)

**Name:** Improper Restriction of Operations within the Bounds of a Memory Buffer

**Description:** The product performs operations on a memory buffer, but it can read from or write to a memory location that is outside of the intended boundary of the buffer.

**Rank:** 18 **ID:** [CWE-798](https://cwe.mitre.org/data/definitions/798.html)

**Name:** Use of Hard-coded Credentials

**Description:** The product contains hard-coded credentials, such as a password or cryptographic key, which it uses for its own inbound authentication, outbound communication to external components, or encryption of internal data.

**Rank:** 19 **ID:** [CWE-918](https://cwe.mitre.org/data/definitions/918.html)

**Name:** Server-Side Request Forgery (SSRF)

**Description:** The web server receives a URL or similar request from an upstream component and retrieves the contents of this URL, but it does not sufficiently ensure that the request is being sent to the expected destination.

**Rank:** 20 **ID:** [CWE-306](https://cwe.mitre.org/data/definitions/306.html)

**Name:** Missing Authentication for Critical Function

**Description:** The product does not perform any authentication for functionality that requires a provable user identity or consumes a significant amount of resources.

**Rank:** 21 **ID:** [CWE-362](https://cwe.mitre.org/data/definitions/362.html)

**Name:** Concurrent Execution using Shared Resources with Improper Synchronization ('Race Condition')

**Description:** The product contains a code sequence that can run concurrently with other code, and the code sequence requires temporary, exclusive access to a shared resource, but a timing window exists in which the shared resource can be modified by another code sequence that is operating concurrently.

**Rank:** 22 **ID:** [CWE-269](https://cwe.mitre.org/data/definitions/269.html)

**Name:** Improper Privilege Management

**Description:** The product does not properly assign, modify, track, or check privileges for an actor, creating an unintended sphere of control for that actor.

**Rank:** 23 **ID:** [CWE-94](https://cwe.mitre.org/data/definitions/94.html)

**Name:** Improper Control of Generation of Code ('Code Injection')

**Description:** The product constructs all or part of a code segment using externally influenced input from an upstream component, but it does not neutralize or incorrectly neutralize special elements that could modify the syntax or behavior of the intended code segment.

**Rank:** 24 **ID:** [CWE-863](https://cwe.mitre.org/data/definitions/863.html)

**Name:** Incorrect Authorization

**Description:** The product performs an authorization check when an actor attempts to access a resource or perform an action, but it does not correctly perform the check. This allows attackers to bypass intended access restrictions.

**Rank:** 25 **ID:** [CWE-276](https://cwe.mitre.org/data/definitions/276.html)

**Name:** Incorrect Default Permissions

**Description:** During installation, installed file permissions are set to allow anyone to modify those files.

Over the years, the list of the**SANS top 25** has undergone considerable changes. One noticeable change is the transition of the list from a more abstract description and version of the weaknesses to a more specific format. The**remapping**and**change** in the rankings of the weaknesses determine their intensity, functionality, and severity.