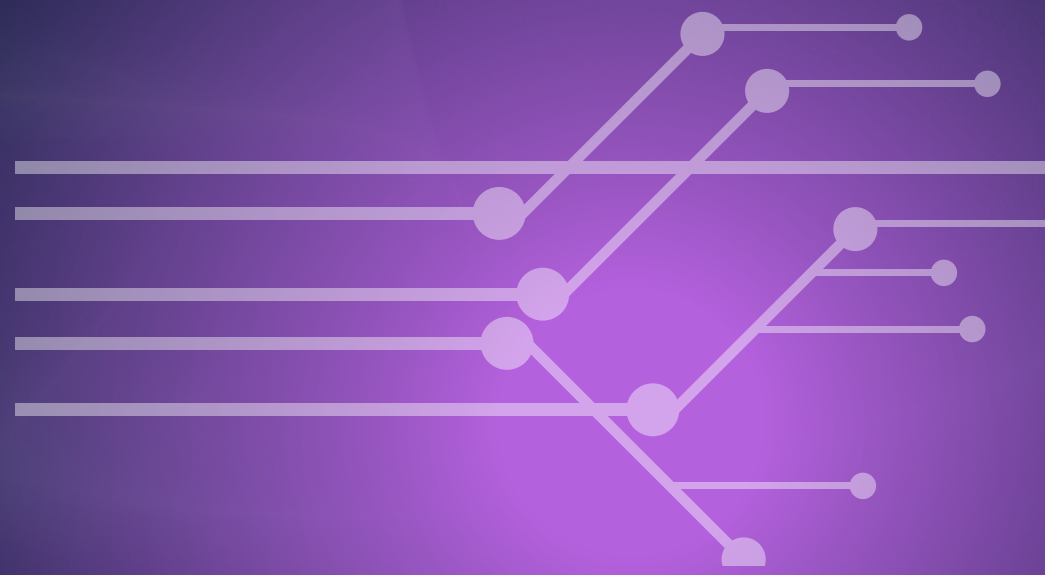
SANS TOP 25

SECURITY V
ULNERABILITIES

CWE/ SANS Top 25



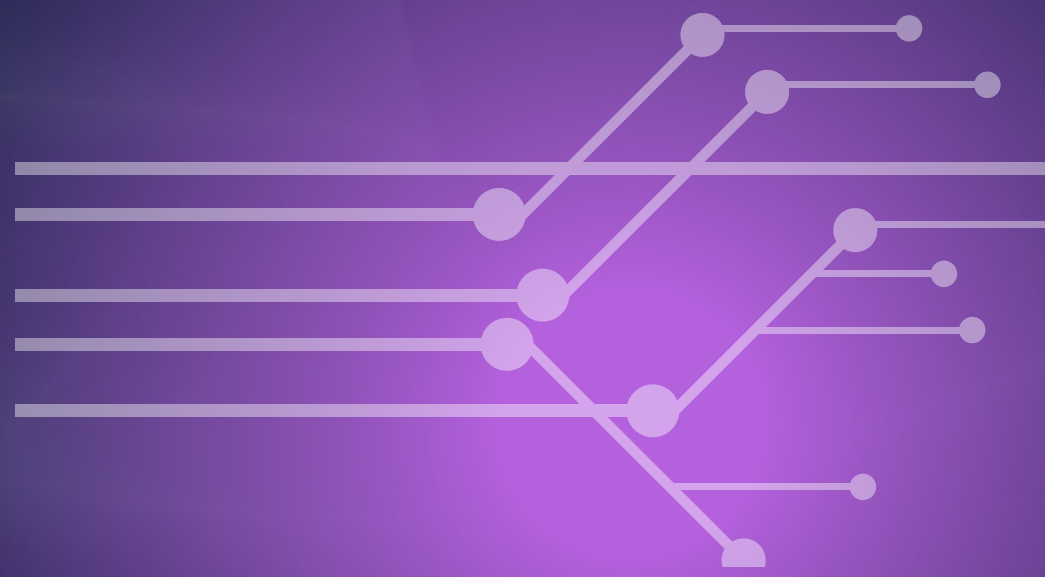
“ The CWE/ SANS top 25 most dangerous software flaws is a list of the most dangerous flaws because they let attackers gain entire control of the software, steal data and information from it, or prohibit it from functioning at all.”

SECURITY V
ULNERABILITIES

CWE/ SANS Top 25

1. OUT-OF-BOUNDS WRITE

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

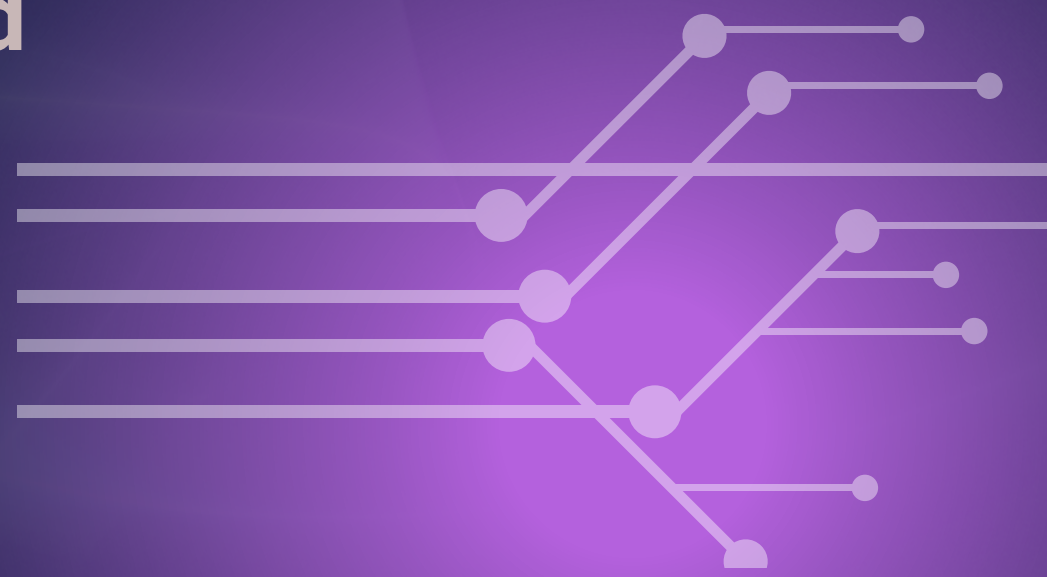


SECURITY V
ULNERABILITIES

CWE/ SANS Top 25

2. IMPROPER NEUTRALIZATION OF INPUT DURING WEB PAGE GENERATION ('CROSS-SITE SCRIPTING')

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



SECURITY V
ULNERABILITIES

CWE/ SANS Top 25

3. MPROPER NEUTRALIZATION OF SPECIAL ELEMENTS USED IN AN SQL COMMAND (‘SQL INJECTION’)

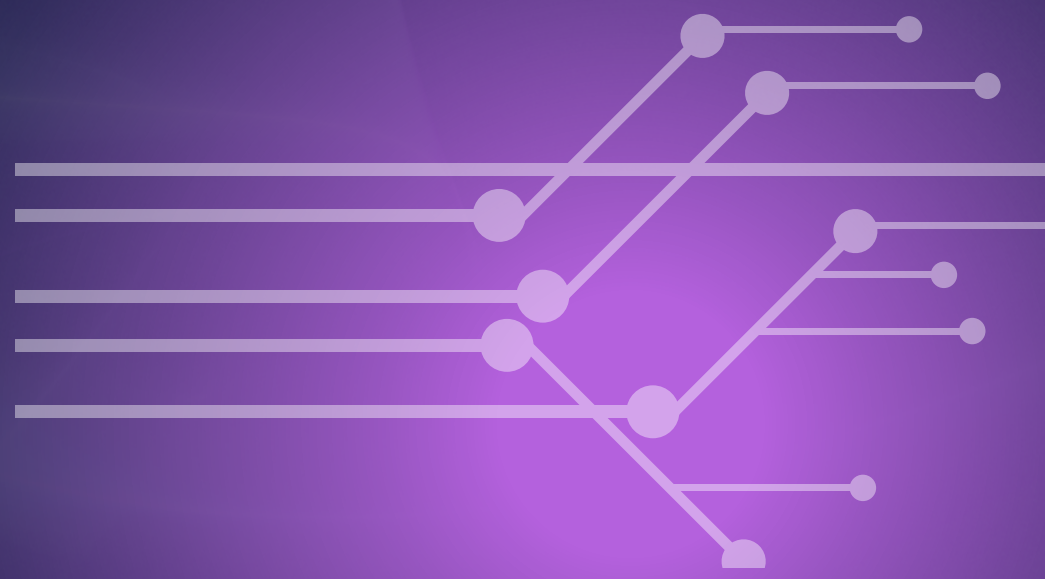
- **The product constructs all or part of an SQL command using externally-influenced input from an upstream component, but it does not neutralize or incorrectly neutralizes special elements that could modify the intended SQL command when it is sent to a downstream component.**

SECURITY V
ULNERABILITIES

CWE/ SANS Top 25

4. USE AFTER FREE

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



SECURITY V
ULNERABILITIES

CWE/ SANS Top 25

5. IMPROPER NEUTRALIZATION OF SPECIAL ELEMENTS USED IN AN OS COMMAND (‘OS COMMAND INJECTION’)

- **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 neutralizes special elements that could modify the intended OS command when it is sent to a downstream component.**

SECURITY V
ULNERABILITIES

CWE/ SANS Top 25

6. OUT-IMPROPER INPUT VALIDATION OF-BOUNDS READ

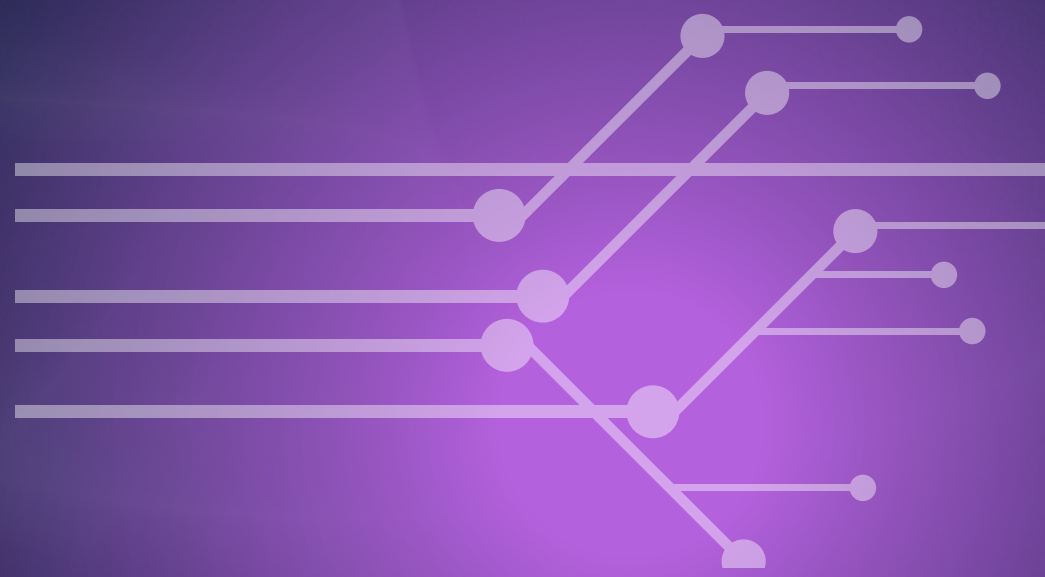
- **The product receives input or data, but it does not validate or incorrectly validates that the input has the properties that are required to process the data safely and correctly.**

SECURITY V
ULNERABILITIES

CWE/ SANS Top 25

7. OUT-OF-BOUNDS READ

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



SECURITY V
ULNERABILITIES

CWE/ SANS Top 25

8. IMPROPER LIMITATION OF A PATH NAME TO A RESTRICTED DIRECTORY (‘PATH TRAVERSAL’)

- **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.**

SECURITY V
ULNERABILITIES

CWE/ SANS Top 25

9. CROSS-SITE REQUEST FORGERY (CSRF)

- **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.**

SECURITY V
ULNERABILITIES

CWE/ SANS Top 25

10. UNRESTRICTED UPLOAD OF FILE WITH DANGEROUS TYPE

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

SECURITY V
ULNERABILITIES

CWE/ SANS Top 25

11. MISSING AUTHORIZATION

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

SECURITY V
ULNERABILITIES

CWE/ SANS Top 25

12. NULL POINTER DEREFERENCE

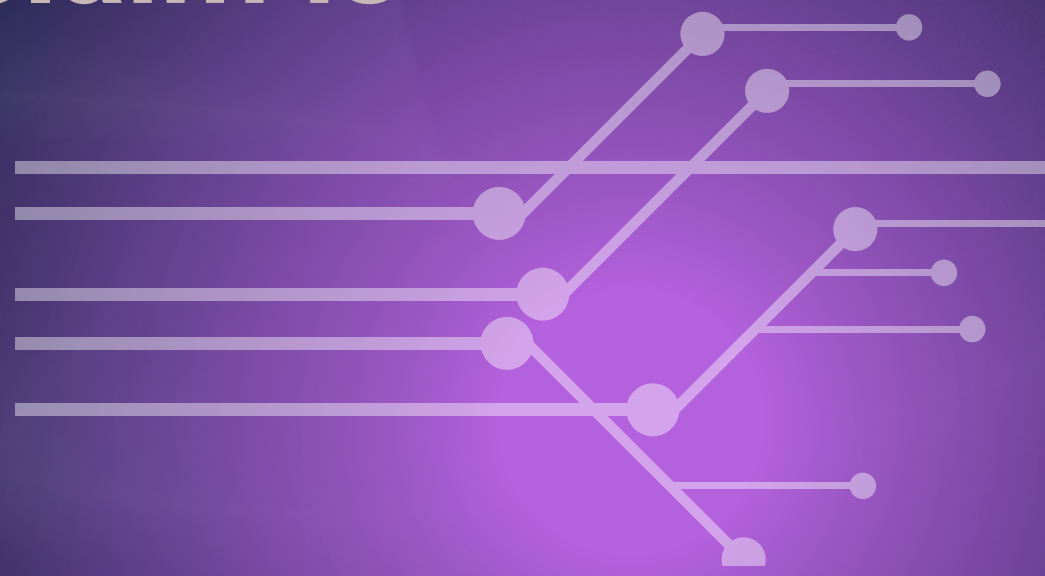
- A NULL pointer dereference occurs when the application dereferences a pointer that it expects to be valid, but is NULL, typically causing a crash or exit.

SECURITY V
ULNERABILITIES

CWE/ SANS Top 25

13. IMPROPER AUTHENTICATION

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



SECURITY V
ULNERABILITIES

CWE/ SANS Top 25

14. IMAGE OVERFLOW OR WRAPAROUND

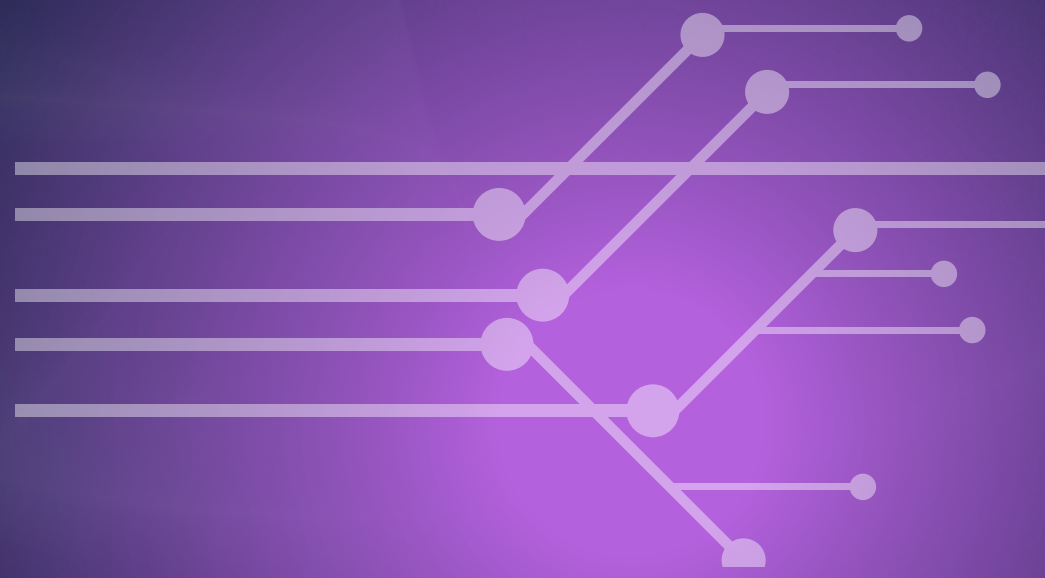
- **The product performs a calculation that can produce an integer overflow or wraparound, when the logic assumes that the resulting value will always be larger than the original value. This can introduce other weaknesses when the calculation is used for resource management or execution control.**
-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

SECURITY V
ULNERABILITIES

CWE/ SANS Top 25

15. DESERIALIZATION OF UNTRUSTED DATA

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



SECURITY V
ULNERABILITIES

CWE/ SANS Top 25

16. IMPROPER NEUTRALIZATION OF SPECIAL ELEMENTS USED IN A COMMAND (‘COMMAND INJECTION’)

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

SECURITY V
ULNERABILITIES

CWE/ SANS Top 25

17. IMPROPER RESTRICTION OF OPERATIONS WITHIN THE BOUNDS OF A MEMORY BUFFER

- **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.**

SECURITY V
ULNERABILITIES

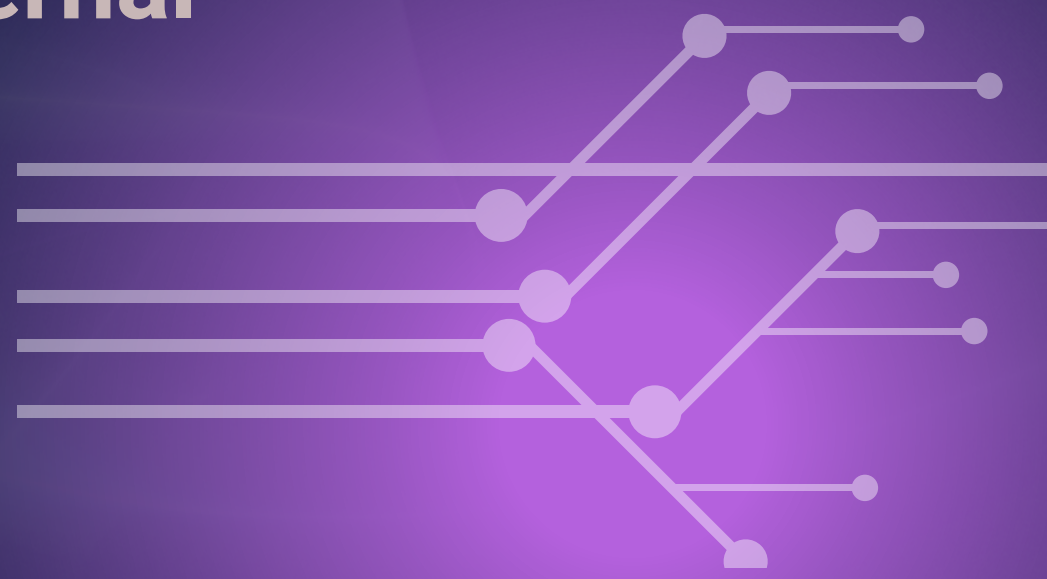
CWE/ SANS Top 25

18. USE OF HARD-CODED CREDENTIALS

- **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.**

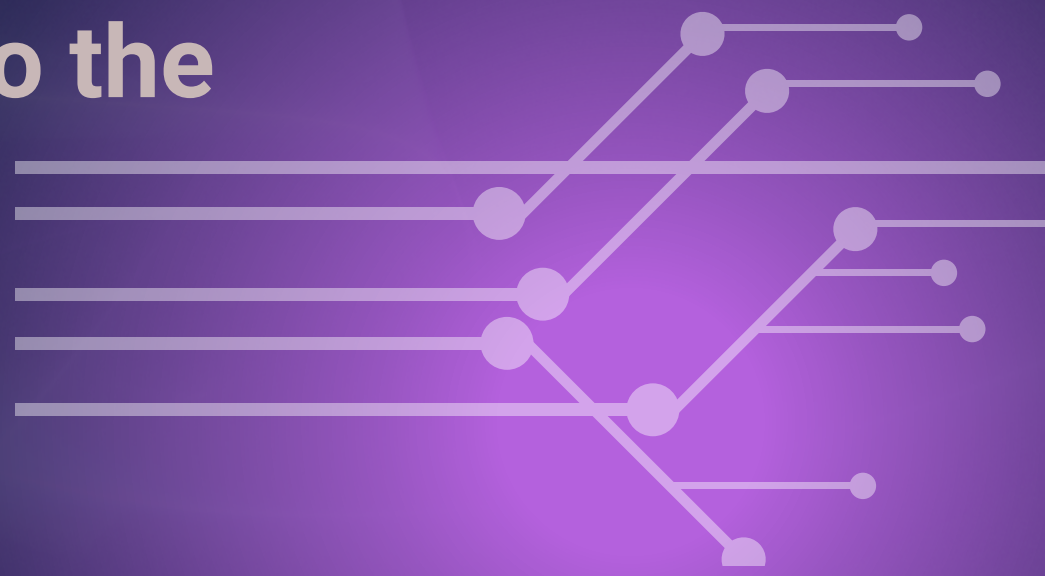
SECURITY V
ULNERABILITIES

CWE/ SANS Top 25



19. SERVER-SIDE REQUEST FORGERY (SSRF)

- 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.



SECURITY V
ULNERABILITIES

CWE/ SANS Top 25

20. MISSING AUTHENTICATION FOR CRITICAL FUNCTION

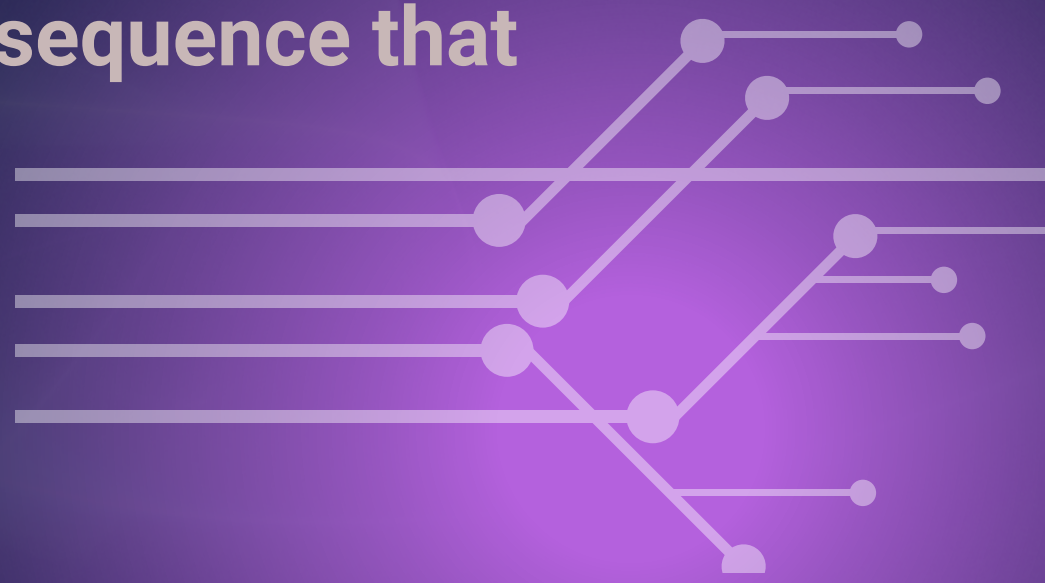
- **The product does not perform any authentication for functionality that requires a provable user identity or consumes a significant amount of resources.**

SECURITY V
ULNERABILITIES

CWE/ SANS Top 25

21. CONCURRENT EXECUTION USING SHARED RESOURCE WITH IMPROPER SYNCHRONIZATION ('RACE CONDITION')

- **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.**



SECURITY V
ULNERABILITIES

CWE/ SANS Top 25

22. IMPROPER PRIVILEGE MANAGEMENT

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

SECURITY V
ULNERABILITIES

CWE/ SANS Top 25

23.IMPROPER CONTROL OF GENERATION OF CODE ('CODE INJECTION')

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

SECURITY V
ULNERABILITIES

CWE/ SANS Top 25

24. INCORRECT AUTHORIZATION

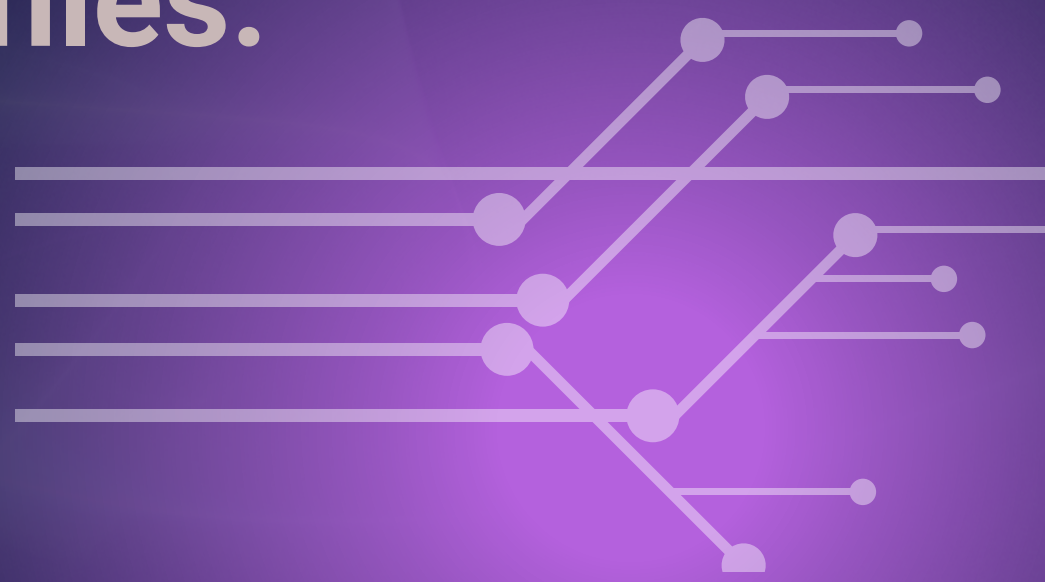
- **The product performs an authorization check when an actor attempts to access a resource or perform an action, but it does not correctly perform the check. This allows attackers to bypass intended access restrictions.**

SECURITY V
ULNERABILITIES

CWE/ SANS Top 25

25. INCORRECT DEFAULT PERMISSIONS

- During installation, installed file permissions are set to allow anyone to modify those files.



SECURITY V
ULNERABILITIES

CWE/ SANS Top 25

DON'T KEEP IT TO YOURSELF – HIT REPOST AND SHARE!

SECURITY V
ULNERABILITIES

CWE/ SANS Top 25