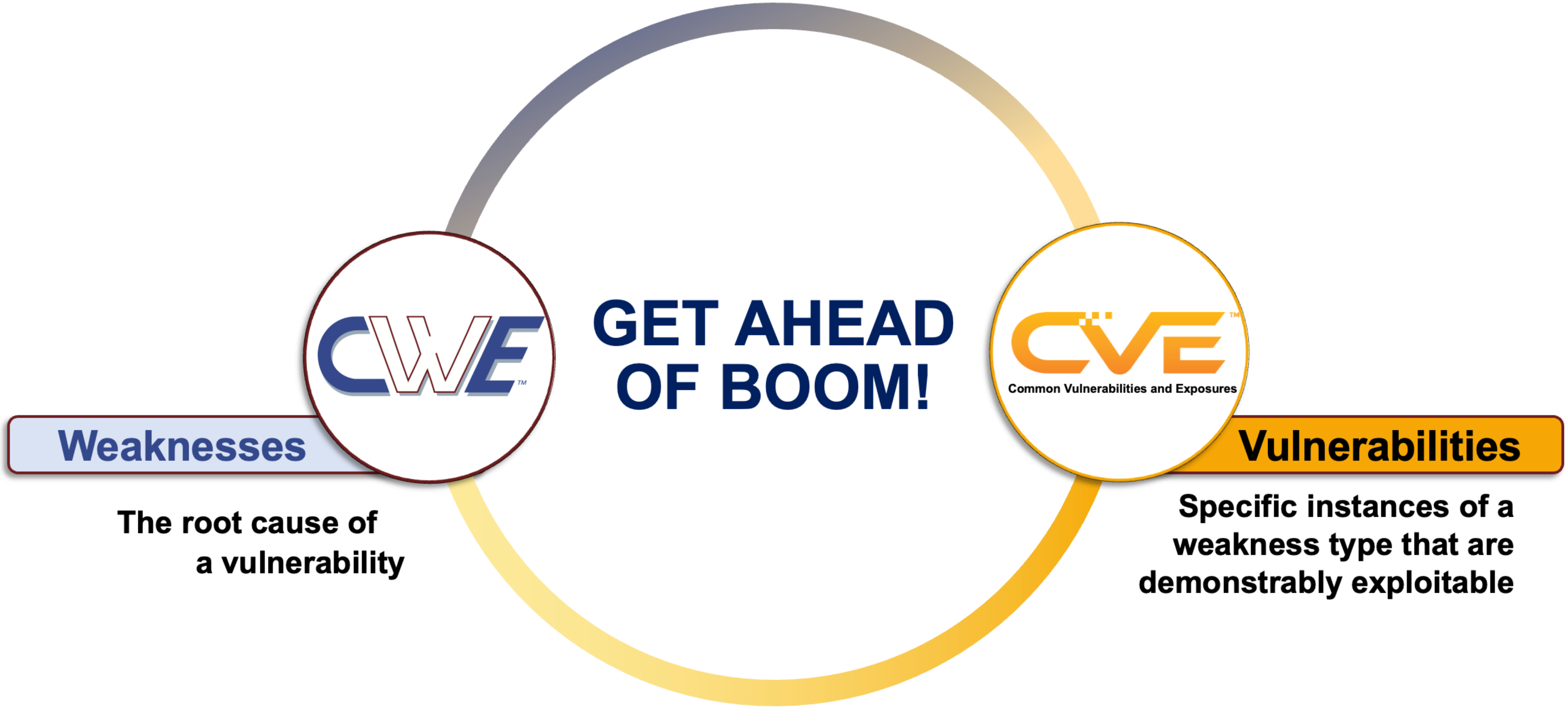
CWE Program Update

Alec Summers

CWE Task Lead

HSSEDI
Homeland Security Systems Engineering & Development Institute™



GET AHEAD OF BOOM!

Weaknesses

The root cause of a vulnerability

Vulnerabilities

Specific instances of a weakness type that are demonstrably exploitable

GET AHEAD OF BOOM!

CWE
Common Vulnerabilities and Exposures
CVE

Weaknesses

CWE-79: Improper Neutralization of Input During Web Page Generation

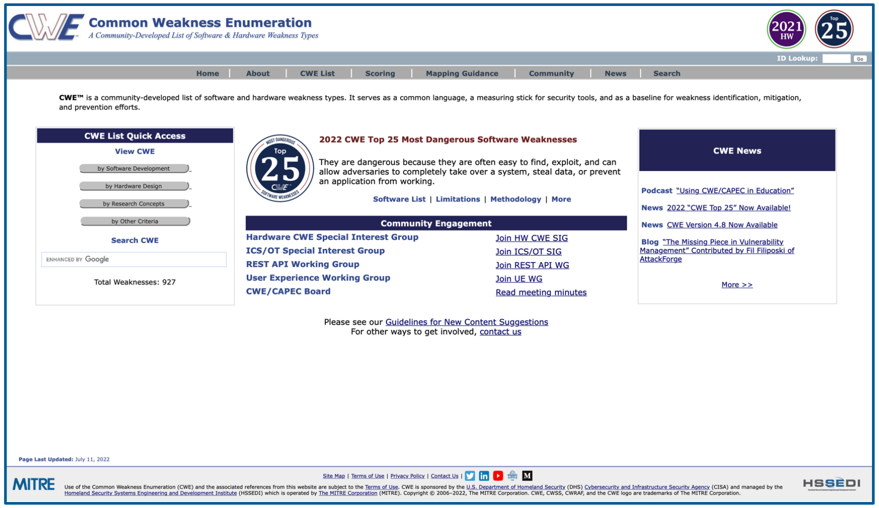
Vulnerabilities

2000+ Cross Site Scripting Injection vulnerabilities in specific technologies in 2022

HSSEDI
Homeland Security Systems Engineering & Development Institute℠

Common Weakness Enumeration (CWE): Helping the Community ‘Get Ahead of Boom!’

- **What CWE is:** A community–developed list of software and hardware weaknesses that allow vulnerabilities to occur
- **CWE value**: Unique identifiers enable understanding of individual root cause conditions behind specific vulnerabilities; enable analysis tools to “speak the same language” on weakness types
- **Value Delivery**: Effective collaboration across stakeholder community in government, industry, and academia
- **MITRE/HSSEDI Role**: Operate the program, maintain the corpus of information, engage the community, manage the vendor compatibility program


CWE
2021 HW
Top 25
ID Lookup:
Go
Home | About | CWE List | Scoring | Mapping Guidance | Community | News | Search
by Software Development
by Hardware Design
by Research Concepts
by Other Criteria
ENHANCED BY Google
Most Dangerous Top 25 CWE Software Weaknesses
Site Map | Terms of Use | Privacy Policy | Contact Us |
MITRE
HSSEDI

Common Weakness Enumeration

A Community-Developed List of Software & Hardware Weakness Types

CWE™ is a community-developed list of software and hardware weakness types. It serves as a common language, a measuring stick for security tools, and as a baseline for weakness identification, mitigation, and prevention efforts.

CWE List Quick Access

View CWE

Search CWE

Total Weaknesses: 927

2022 CWE Top 25 Most Dangerous Software Weaknesses

They are dangerous because they are often easy to find, exploit, and can allow adversaries to completely take over a system, steal data, or prevent an application from working.

Software List | **Limitations** | **Methodology** | **More**

Community Engagement

Hardware CWE Special Interest Group	Join HW CWE SIG
ICS/OT Special Interest Group	Join ICS/OT SIG
REST API Working Group	Join REST API WG
User Experience Working Group	Join UE WG
CWE/CAPEC Board	Read meeting minutes

CWE News

Podcast “Using CWE/CAPEC in Education”

News 2022 “CWE Top 25” Now Available!

News CWE Version 4.8 Now Available

Blog “The Missing Piece in Vulnerability Management” Contributed by Fil Filiposki of AttackForge

More >>

Please see our Guidelines for New Content Suggestions
For other ways to get involved, contact us

Page Last Updated: July 11, 2022

Use of the Common Weakness Enumeration (CWE) and the associated references from this website are subject to the Terms of Use. CWE is sponsored by the U.S. Department of Homeland Security (DHS) Cybersecurity and Infrastructure Security Agency (CISA) and managed by the Homeland Security Systems Engineering and Development Institute (HSSEDI) which is operated by The MITRE Corporation (MITRE). Copyright © 2006–2022, The MITRE Corporation. CWE, CWSS, CWRAF, and the CWE logo are trademarks of The MITRE Corporation.

933 Total Weaknesses

HSSEDI is a trademark of the U.S. Department of Homeland Security (DHS).
The HSSEDI FFRDC is managed and operated by The MITRE Corporation for DHS.

CWE Program

Mission

Identify, define, and catalog known weakness types in software, firmware, hardware, or service components

Goals

1. Increase program adoption
   - ➢ Expand use of CWE in the vulnerability management ecosystem
2. Increase program coverage
   - ➢ More weakness types are identified and defined so that they can be understood and avoided

Desired Outcome

Products are more secure because weaknesses are eliminated or avoided, thereby thwarting attackers

The Old CWE Program!

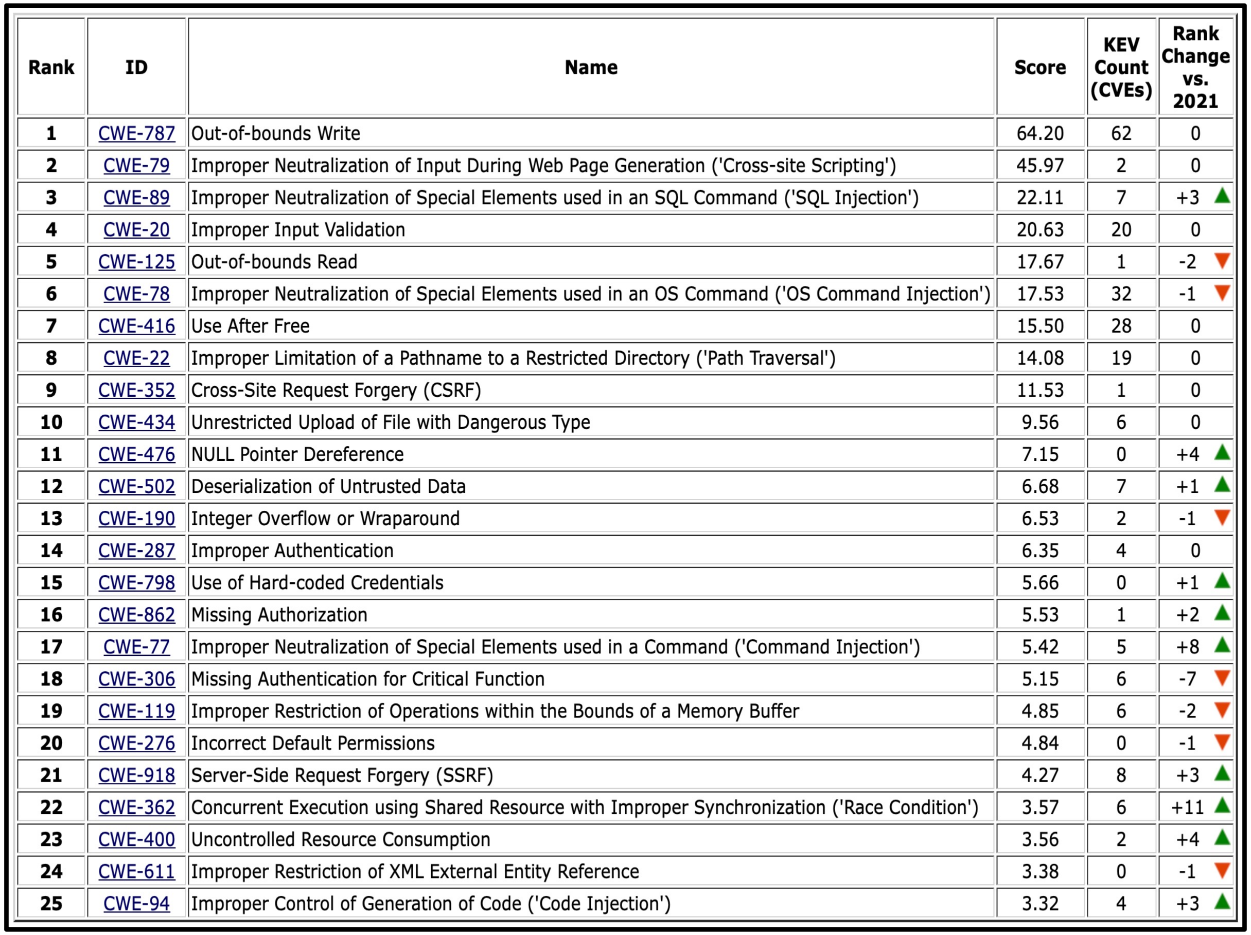
- **MITRE produced all CWE content updates and new entries virtually alone, with minimal and inconsistent external engagement**

- **This model limits the ability to execute effectively against our mission and goals**
  - Slow entry development
  - Technical debt builds as updates take more time
  - Expertise limited to CWE team, limits content expansion to new domain areas
  - Too much separation between users and program operators

2019/2020: Shift in Strategy

- **Engage the CWE stakeholder community**
  - Establish a Board of key stakeholders across industry, government, and academia
  - Stand up working groups and special interest groups to collaboratively work on key issues
- **Define that path forward**
  - Work with stakeholders to expand content development efforts to new domains (e.g., hardware)
  - Identify ways user experience can improve and work with the community to address them
- **Execute and adjust**
  - Keep doing what works, stop doing what does not

CWE Top 25 Most Dangerous Software Weaknesses

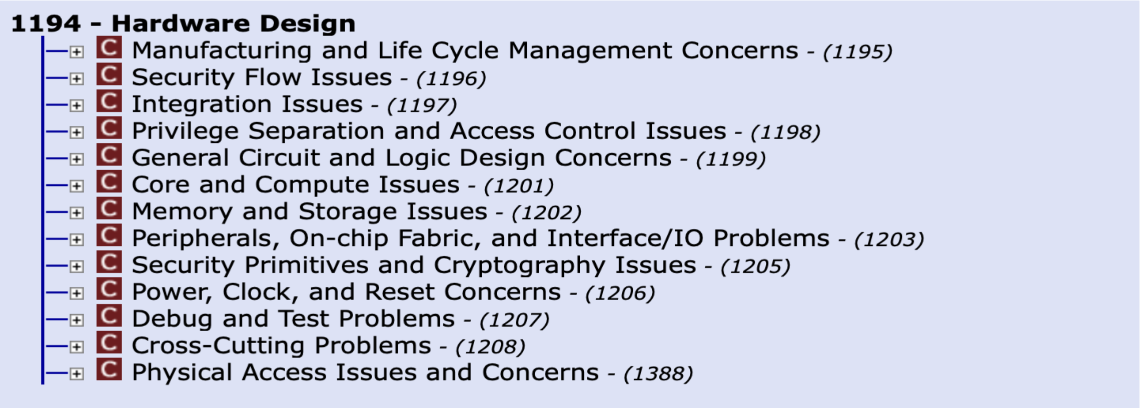


Rank	ID	Name	Score	KEV Count (CVEs)	Rank Change vs. 2021
1	CWE-787	Out-of-bounds Write	64.20	62	0
2	CWE-79	Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')	45.97	2	0
3	CWE-89	Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')	22.11	7	+3 ▲
4	CWE-20	Improper Input Validation	20.63	20	0
5	CWE-125	Out-of-bounds Read	17.67	1	-2 ▼
6	CWE-78	Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')	17.53	32	-1 ▼
7	CWE-416	Use After Free	15.50	28	0
8	CWE-22	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal')	14.08	19	0
9	CWE-352	Cross-Site Request Forgery (CSRF)	11.53	1	0
10	CWE-434	Unrestricted Upload of File with Dangerous Type	9.56	6	0
11	CWE-476	NULL Pointer Dereference	7.15	0	+4 ▲
12	CWE-502	Deserialization of Untrusted Data	6.68	7	+1 ▲
13	CWE-190	Integer Overflow or Wraparound	6.53	2	-1 ▼
14	CWE-287	Improper Authentication	6.35	4	0
15	CWE-798	Use of Hard-coded Credentials	5.66	0	+1 ▲
16	CWE-862	Missing Authorization	5.53	1	+2 ▲
17	CWE-77	Improper Neutralization of Special Elements used in a Command ('Command Injection')	5.42	5	+8 ▲
18	CWE-306	Missing Authentication for Critical Function	5.15	6	-7 ▼
19	CWE-119	Improper Restriction of Operations within the Bounds of a Memory Buffer	4.85	6	-2 ▼
20	CWE-276	Incorrect Default Permissions	4.84	0	-1 ▼
21	CWE-918	Server-Side Request Forgery (SSRF)	4.27	8	+3 ▲
22	CWE-362	Concurrent Execution using Shared Resource with Improper Synchronization ('Race Condition')	3.57	6	+11 ▲
23	CWE-400	Uncontrolled Resource Consumption	3.56	2	+4 ▲
24	CWE-611	Improper Restriction of XML External Entity Reference	3.38	0	-1 ▼
25	CWE-94	Improper Control of Generation of Code ('Code Injection')	3.32	4	+3 ▲

- **Published annually**
- **Analysis of NIST National Vulnerability Database over the previous two calendar years**
- **Methodology focuses on weakness type prevalence and severity (based on CVSS scores)**
- **2022 list included CISA Known Exploited Vulnerability information**

CWE Scope Expansion into Hardware Domain

- **Expansion of scope to include Hardware Design Weaknesses**
  - From 2006 to 2019, CWE's scope was strictly software
  - In February 2020, v4.0 expanded into Hardware Design weakness content
  - CWE v4.1 – v4.9 (most recently in October 2022)
  - 100 HW-related entries broken down into 13 categories



1194 - Hardware Design

- ⊞ C Manufacturing and Life Cycle Management Concerns - *(1195)*
- ⊞ C Security Flow Issues - *(1196)*
- ⊞ C Integration Issues - *(1197)*
- ⊞ C Privilege Separation and Access Control Issues - *(1198)*
- ⊞ C General Circuit and Logic Design Concerns - *(1199)*
- ⊞ C Core and Compute Issues - *(1201)*
- ⊞ C Memory and Storage Issues - *(1202)*
- ⊞ C Peripherals, On-chip Fabric, and Interface/IO Problems - *(1203)*
- ⊞ C Security Primitives and Cryptography Issues - *(1205)*
- ⊞ C Power, Clock, and Reset Concerns - *(1206)*
- ⊞ C Debug and Test Problems - *(1207)*
- ⊞ C Cross-Cutting Problems - *(1208)*
- ⊞ C Physical Access Issues and Concerns - *(1388)*

MOST IMPORTANT
2021
CWE™
HARDWARE WEAKNESSES

Community Engagement

Working with the community to expand coverage and drive adoption

- **CWE Board**
  - 14 individuals representing organizations across government, industry, and academia to set/promote the goals/objectives of the CWE/CAPEC Program to ensure the ongoing adoption, coverage, and quality of both corpuses
- **Hardware CWE Special Interest Group**
  - 129 participants, ~25 per session
- **User Experience Working Group**
  - 77 participants, ~15 per session
- **REST API Working Group**
  - 42 participants, ~20 per session
- **ICS/OT Special Interest Group (and 2 sub-working groups)**
  - 183 participants, ~40 per session

Current Activities and Focus Areas

- **CWE v4.10 coming January 31**
  - New and improved weakness entries in ICS, HW
- **HW CWE SIG**
  - Current research discussions on formalizing CWE’s coverage of transient execution weaknesses
- **Research**
  - Weakness grouping trend analysis (e.g., memory safety, injection, access control)
- **Working with the community to improve weakness mapping throughout the vulnerability management ecosystem**
  - e.g., Improving CWE usability, developing training/guidance materials

Wrap-Up

For more information, please contact cwe@mitre.org

Twitter, @CweCapec

Blogs via Medium, @CWE_CAPEC

LinkedIn, CVE-CWE-CAPEC

Alec J. Summers

asummers@mitre.org

MITRE
Principal Cyber Security Engineer
Group Lead, Cybersecurity Operations and Integration

HSSEDI
Homeland Security Systems Engineering & Development Institute™