2021 CWE Top 25 Most Dangerous Software Weaknesses

Introduction

The 2021 Common Weakness Enumeration (CWE™) Top 25 Most Dangerous Software Weaknesses (CWE Top 25) is a demonstrative list of the most common and impactful issues experienced over the previous two calendar years. These weaknesses 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. The CWE Top 25 is a valuable community resource that can help developers, testers, and users — as well as project managers, security researchers, and educators — provide insight into the most severe and current security weaknesses.

To create the 2021 list, the CWE Team leveraged Common Vulnerabilities and Exposures (CVE®) data found within the National Institute of Standards and Technology (NIST) National Vulnerability Database (NVD), as well as the Common Vulnerability Scoring System (CVSS) scores associated with each CVE record. A formula was applied to the data to score each weakness based on prevalence and severity.

The CWE Top 25

Below is a brief listing of the weaknesses in the 2021 CWE Top 25, including the overall score of each.

Rank	ID	Name	Score	2020 Rank Change
[1]	CWE-787	Out-of-bounds Write	65.93	+1
[2]	CWE-79	Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')	46.84	-1
[3]	CWE-125	Out-of-bounds Read	24.9	+1
[4]	CWE-20	Improper Input Validation	20.47	-1
[5]	CWE-78	Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')	19.55	+5
[6]	CWE-89	Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')	19.54	0
[7]	CWE-416	Use After Free	16.83	+1
[8]	CWE-22	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal')	14.69	+4
[9]	CWE-352	Cross-Site Request Forgery (CSRF)	14.46	0
[10]	CWE-434	Unrestricted Upload of File with Dangerous Type	8.45	+5
[11]	CWE-306	Missing Authentication for Critical Function	7.93	+13
[12]	CWE-190	Integer Overflow or Wraparound	7.12	-1
[13]	CWE-502	Deserialization of Untrusted Data	6.71	+8
[14]	CWE-287	Improper Authentication	6.58	0
[15]	CWE-476	NULL Pointer Dereference	6.54	-2
[16]	CWE-798	Use of Hard-coded Credentials	6.27	+4
[17]	CWE-119	Improper Restriction of Operations within the Bounds of a Memory Buffer	5.84	-12
[18]	CWE-862	Missing Authorization	5.47	+7
[19]	CWE-276	Incorrect Default Permissions	5.09	+22
[20]	CWE-200	Exposure of Sensitive Information to an Unauthorized Actor	4.74	-13
[21]	CWE-522	Insufficiently Protected Credentials	4.21	-3
[22]	CWE-732	Incorrect Permission Assignment for Critical Resource	4.2	-6

Rank	ID	Name	Score	2020 Rank Change
[23]	CWE-611	Improper Restriction of XML External Entity Reference	4.02	-4
[24]	CWE-918	Server-Side Request Forgery (SSRF)	3.78	+3
[25]	CWE-77	Improper Neutralization of Special Elements used in a Command ('Command Injection')	3.58	+6

Analysis and Comment

The major difference between the 2020 and 2021 CWE Top 25 lists is the continued transition to more specific weaknesses as opposed to abstract, class-level weaknesses. A preliminary estimate suggests that the percentage of Base-level CWEs has increased from ~60% to ~71% of all Top 25 entries, and the percentage of Class-level CWEs has decreased from ~30% to ~20% of entries. Other weakness levels (e.g., category, compound, and variant) remain relatively unchanged.

While a few class-level weaknesses still exist in the list, they have declined noticeably in the ranking, as influenced by prioritization in the remapping task (see Remapping Task section below). This movement is expected to continue in future years as the community improves its mappings to more precise weaknesses.

With the relative decline of class-level weaknesses, more specific CWEs have moved up to take the place of these high-level classes, such as CWE-78 (Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')), CWE-22 (Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal')), CWE-434 (Unrestricted Upload of File with Dangerous Type), CWE-306 (Missing Authentication for Critical Function), CWE-502 (Deserialization of Untrusted Data), CWE-862 (Missing Authorization), and CWE-276 (Incorrect Default Permissions). Subsequent future movement will greatly benefit users that are attempting to understand the actual issues that threaten today’s systems, as the Top 25 Team believes that Base-level weaknesses are more informative to stakeholders than Class-level weaknesses.

The biggest movement up the list is:

- CWE-276 (Incorrect Default Permissions): from #41 to #19
- CWE-306 (Missing Authentication for Critical Function): from #24 to #11
- CWE-502 (Deserialization of Untrusted Data): from #21 to #13

- CWE-862 (Missing Authorization): from #25 to #18
- CWE-77 (Improper Neutralization of Special Elements used in a Command ('Command Injection')): from #31 to #25

Most of these weaknesses represent some of the most difficult areas to analyze a system on. A theory about this movement is that the community has improved its education, tooling, and analysis capabilities related to some of the more implementation specific weaknesses identified in previous editions of the CWE Top 25 and have reduced their occurrence. This would lower their ranking, in turn raising the ranking of these more difficult weaknesses.

Five of the biggest downward movers are:

- CWE-200 (Exposure of Sensitive Information to an Unauthorized Actor): from #7 to #20
- CWE-119 (Improper Restriction of Operations within the Bounds of a Memory Buffer): from #5 to #17
- CWE-94 (Improper Control of Generation of Code ('Code Injection')): from #17 to #28
- CWE-269 (Improper Privilege Management): from #22 to #29
- CWE-732 (Incorrect Permission Assignment for Critical Resource): from #16 to #22

New entries in the Top 25 are:

- CWE-276 (Incorrect Default Permissions): from #41 to #19
- CWE-918 (Server-Side Request Forgery (SSRF)): from #27 to #24
- CWE-77 (Improper Neutralization of Special Elements used in a Command ('Command Injection')): from #31 to #25

Entries that fell off the Top 25 are:

- CWE-400 (Uncontrolled Resource Consumption): from #23 to #27
- CWE-94 (Improper Control of Generation of Code ('Code Injection')): from #17 to #28
- CWE-269 (Improper Privilege Management): from #22 to #29

For those who are interested in why these shifts happened, see the "Remapping Task" section to learn how prioritization of remapping activities may have affected the rankings.

Methodology

The 2021 CWE Top 25 was developed by obtaining published vulnerability data from the NVD. The NVD obtains vulnerability data from CVE and then supplements it with additional analysis and information including a mapping to one

or more weaknesses, and a CVSS score, which is a numerical score representing the potential severity of a vulnerability based upon a standardized set of characteristics about the vulnerability. NVD provides this information in a digestible format that is used for the data-driven approach in creating the 2021 CWE Top 25. This approach provides an objective look at what vulnerabilities are currently seen in the real world, creates a foundation of analytical rigor built on publicly reported vulnerabilities instead of subjective surveys and opinions, and makes the process easily repeatable.

The 2021 CWE Top 25 leverages NVD data with CVE IDs from the years 2019 and 2020, as downloaded on March 18, 2021. This snapshot of raw data consists of approximately 32,500 CVEs that are associated with a weakness.

A scoring formula is used to calculate a ranked order of weaknesses that combines the frequency that a CWE is the root cause of a vulnerability with the projected severity of its exploitation. In both cases, the frequency and severity are normalized relative to the minimum and maximum values seen.

To determine a CWE's frequency, the scoring formula calculates the number of times a CWE is mapped to a CVE within the NVD. Only those CVEs that have an associated weakness are used in this calculation, since using the entire set of CVEs within the NVD would result in very low frequency rates and very little difference amongst the different weakness types.

Freq = {count(CWE_X' ∈ NVD) for each CWE_X' in NVD}

Fr(CWE_X) = (count(CWE_X ∈ NVD) - min(Freq)) / (max(Freq) - min(Freq))

The other component in the scoring formula is a weakness' severity, which is represented by the average CVSS score of all CVEs that map to the particular CWE. The equation below is used to calculate this value.

Sv(CWE_X) = (average_CVSS_for_CWE_X - min(CVSS)) / (max(CVSS) - min(CVSS))

The level of danger presented by a particular CWE is then determined by multiplying the severity score by the frequency score.

Score(CWE_X) = Fr(CWE_X) * Sv(CWE_X) * 100

There are a few properties of the methodology that merit further explanation.

- Weaknesses that are rarely discovered will not receive a high score, regardless of the typical consequence associated with any exploitation. This makes sense, since if developers are not making a particular mistake, then the weakness should not be highlighted in the CWE Top 25.
- Weaknesses with a low impact will not receive a high score. This again makes sense, since the inability to cause significant harm by exploiting a weakness means that weakness should be ranked below those that can.

- Weaknesses that are both common and can cause significant harm should receive a high score.

The CWE Top 25 with Scoring Metrics

The following table shows the 2021 CWE Top 25 with relevant scoring information, including the number of entries related to a particular CWE within the NVD data set, and the average CVSS score for each weakness.

Rank	CWE	NVD Count	Avg CVSS	Overall Score
[1]	CWE-787	3033	8.22	65.93
[2]	CWE-79	3564	5.80	46.84
[3]	CWE-125	1448	6.94	24.90
[4]	CWE-20	1120	7.25	20.47
[5]	CWE-78	833	8.71	19.55
[6]	CWE-89	830	8.73	19.54
[7]	CWE-416	807	7.98	16.83
[8]	CWE-22	783	7.39	14.69
[9]	CWE-352	741	7.60	14.46
[10]	CWE-434	381	8.36	8.45
[11]	CWE-306	381	7.98	7.93
[12]	CWE-190	368	7.56	7.12
[13]	CWE-502	280	8.87	6.71
[14]	CWE-287	324	7.84	6.58
[15]	CWE-476	404	6.67	6.54
[16]	CWE-798	275	8.54	6.27
[17]	CWE-119	278	8.04	5.84
[18]	CWE-862	361	6.38	5.47
[19]	CWE-276	298	6.92	5.09

Rank	CWE	NVD Count	Avg CVSS	Overall Score
[20]	CWE-200	330	6.16	4.74
[21]	CWE-522	232	7.23	4.21
[22]	CWE-732	249	6.87	4.20
[23]	CWE-611	206	7.62	4.02
[24]	CWE-918	207	7.26	3.78
[25]	CWE-77	164	8.28	3.58

Weaknesses On the Cusp

Continuing on the theme from last year, the CWE team feels it is important to share these fifteen additional weaknesses that scored just outside of the final Top 25. Per the scoring formula, these weaknesses were potentially not severe enough, or not prevalent enough, to be included in the 2021 CWE Top 25.

Individuals that perform mitigation and risk decision-making using the 2021 CWE Top 25 may want to consider including these additional weaknesses in their analyses:

Rank	CWE	Name	NVD Count	Avg CVSS	Overall Score	2020 Rank Change
[26]	CWE-295	Improper Certificate Validation	201	6.99	3.47	+2
[27]	CWE-400	Uncontrolled Resource Consumption	200	6.99	3.46	-4
[28]	CWE-94	Improper Control of Generation of Code ('Code Injection')	138	8.63	3.18	-11
[29]	CWE-269	Improper Privilege Management	172	7.30	3.16	-7
[30]	CWE-917	Improper Neutralization of Special Elements used in an Expression Language Statement ('Expression Language Injection')	128	9.05	3.14	+17

Rank	CWE	Name	NVD Count	Avg CVSS	Overall Score	2020 Rank Change
[31]	CWE-59	Improper Link Resolution Before File Access ('Link Following')	177	7.11	3.13	+9
[32]	CWE-401	Missing Release of Memory after Effective Lifetime	205	6.42	3.13	0
[33]	CWE-362	Concurrent Execution using Shared Resource with Improper Synchronization ('Race Condition')	202	6.47	3.12	+1
[34]	CWE-427	Uncontrolled Search Path Element	158	7.66	3.10	+24
[35]	CWE-319	Cleartext Transmission of Sensitive Information	177	6.99	3.06	+7
[36]	CWE-843	Access of Resource Using Incompatible Type ('Type Confusion')	124	8.51	2.80	+9
[37]	CWE-601	URL Redirection to Untrusted Site ('Open Redirect')	184	6.12	2.62	-2
[38]	CWE-863	Incorrect Authorization	155	6.80	2.57	-9
[39]	CWE-532	Insertion of Sensitive Information into Log File	171	6.26	2.51	-6
[40]	CWE-770	Allocation of Resources Without Limits or Throttling	136	6.99	2.34	-1

Limitations of the Methodology

There are several limitations to the data-driven approach used in creating the CWE Top 25.

Data Bias

First, the approach only uses data that was publicly reported and captured in the NVD, and numerous vulnerabilities exist that do not have CVE IDs. Vulnerabilities that are not included in the NVD are therefore excluded from this approach. For

example, CVE/NVD typically does not cover vulnerabilities found and fixed before any system has been publicly released, in online services, or in bespoke software that is internal to a single organization. Weaknesses that lead to these types of vulnerabilities may be under-represented in the 2021 CWE Top 25.

Second, even for vulnerabilities that receive a CVE, often there is not enough information to make an accurate (or precise) identification of the appropriate CWE being exploited. Many CVE entries are published by vendors who only describe the impact of the vulnerability without providing details of the vulnerability itself. For example, over 4,800 CVEs from 2019 and 2020 did not have sufficient information to determine the underlying weakness. In other cases, the CVE description covers how the vulnerability is attacked – but this does not always indicate what the associated weakness is. For example, if a long input to a program causes a crash, the cause of the crash could be due to a buffer overflow, a reachable assertion, excessive memory allocation, an unhandled exception, etc. These all correspond to different, individual CWEs. In other CVE entries, only generic terms are used such as "malicious input," which gives no indication of the associated weakness. For some entries, there may be useful information available in the references, but it is difficult to analyze. For example, a researcher might use a fuzzing program that generates a useful test case that causes a crash, but the developer simply fixes the crash without classifying and reporting what the underlying mistake was.

Third, there is inherent bias in the CVE/NVD dataset due to the set of vendors that report vulnerabilities and the languages that are used by those vendors. If one of the largest contributors to CVE/NVD primarily uses C as its programming language, the weaknesses that often exist in C programs are more likely to appear. Fuzzing programs can be very effective against memory-based programs, so they may find many more vulnerabilities. The scoring metric outlined above attempts to mitigate this bias by looking at more than just the most frequently reported CWEs; it also takes into consideration average CVSS score.

Another bias in the CVE/NVD dataset is that most vulnerability researchers and/or detection tools are very proficient at finding certain weaknesses but not others. Those types of weakness that researchers and tools struggle to find will end up being under-represented within the 2021 CWE Top 25.

Finally, gaps or suspected mischaracterizations of the CWE hierarchy itself lead to incorrect mappings. The ongoing remapping work helps the CWE Team learn about these content gaps and issues, which will be addressed in subsequent CWE releases.

Metric Bias

An important bias to understand related to the metric is that it indirectly prioritizes implementation flaws over design flaws, due to their prevalence within individual software packages. For example, a web application may have many different cross-site scripting (XSS) vulnerabilities due to large attack surface, yet only one instance of weak authentication that could compromise the entire application. An alternate metric could be devised that includes the

percentage of products within NVD that have at least one CVE with a particular CWE. This kind of metric is often used by application security vendors in their annual analyses.

Another limitation of the metric was raised in December 2020 by Galhardo, Bojanova, Mell, and Gueye in their ACSC paper "Measurements of the Most Significant Software Security Weaknesses". The authors "find that the published equation highly biases frequency and almost ignores exploitability and impact in generating top lists of varying sizes. This is due to the differences in the distributions of the component metric values." That critique seems to apply in this year's list as well. For example, consider how CWE-79 is ranked #2, but it has the lowest average CVSS score (5.80) of the entire Top 25 and the Cusp.

Over the next year, the Top 25 Team will actively investigate alternate metrics such as these, and a new metric might be selected to determine the 2022 CWE Top 25.

Remapping Task

To prepare the CVE/NVD data for analysis, the CWE Team reviewed the CWE mappings of selected CVE/NVD entries and, where appropriate, "remapped" the entries so that they referenced more appropriate CWE IDs.

This remapping work was performed on over nine thousand CVE entries in consideration for the 2021 Top 25 List. The remapped data has been shared with NIST so that they can update their CVE entries within NVD.

The primary activities were:

- Download a "snapshot" of NVD data from 2019 and 2020. This repository was downloaded on March 18, 2021, and it was used throughout analysis.
- Perform automated keyword searches to find likely remaps to CWE entries that were incorrectly mapped. For example, some CVE entries were mapped to the higher-level CWE-119 (Improper Restriction of Operations within the Bounds of a Memory Buffer). However, phrases related to out-of-bounds read were automatically discoverable within CVE descriptions. In those cases, mapping to the lower level CWE-125 (Out-of-bounds Read) is considered more appropriate.
- Extend the keyword matcher to detect potential mappings for many more CWE entries, including some that had only been created in the past year or two. A related matcher was created to look for actual names of CWE entries within CVE descriptions, which was surprisingly successful.
- Identify the four highest-ranking classes based on a snapshot calculation of the Top 25 using the original March 18 NVD data, then investigate them more closely. These "top 4" focused classes were CWE-20, CWE-200, CWE-119, and CWE-269.

- Choose additional subject areas for emphasis. This year's emphasis included access control, cryptography, and randomness. Note that this is the first year in which cryptography-related CVEs were analyzed as a group; although this did not affect the Top 25, it was highly useful for understanding limitations of CWE for supporting mapping with respect to cryptography. Access control was also targeted because of the prevalence of class-level access-control issues in the Top 25 and the Cusp, as well as a suspicion that mappings would be inconsistent, which turned out to be true.
- Perform deeper analysis in some areas suspected to have mapping inaccuracies, especially injection. This typically involved looking more closely at references than in past years. For example, based on analysis in 2020, mappings to CWE-74 and CWE-77 were already known to have a good chance of being mapped to the more-precise CWE-78 when analyzing references. (https://medium.com/@CWE_CAPEC/2020-cwe-top-25-analysis-c39d100cb0fd)
- De-prioritize categories. There were few CVEs remaining that still mapped to categories, due to the elimination of categories in View 1003 in late 2019. CVEs mapped to categories were not analyzed as a group per se, but they were often included in other groups such as cryptography or keyword matches. It should be noted that some CNA sources still used categories in 2020 data.
- Improve the manual mapping process with automated tooling and annotations. While some of these improvements were experimental, they are likely to be used in future Top 25 lists, such as automated syntax checks for remapping reports provided by analysts; the automated scraping of reference URLs for CWE IDs and keyword matches; and the shifting of CVE records between different "analysis batches" to deprioritize or reassign CVEs that presented more complex analysis challenges.
- While the CWE team made every possible effort to minimize subjectivity in the remapping corrections, the lack of relevant, detailed information present in some CVE descriptions meant that a small portion of the dataset still required some subjective analytical conclusions.

In previous years, at the same time as the Top 25 release, the CWE-1003 view was also modified to ensure that it could still provide coverage for the most common CWE mappings. This created additional technical complexity for both NIST and the CWE Top 25 Team. This year, View 1003 will be updated in the CWE 4.6 release, possibly in October.

Limitations of the Remapping Task

After using this remapping methodology for the 2019, 2020, and 2021 Top 25 lists, some limitations have become apparent:

- The number of CVEs with high-level CWE entries remains high, forcing manual remapping of a large number of CVEs, which is labor-intensive.
- Remapping was performed over a short time frame before publication of the list, which increased timing and staffing pressures during this period. Also, data exchange with NIST was changed to provide mapping data over the entire review period, instead of all at once at the end. Still, the short time frame made it difficult for NVD staff to receive, analyze, and process all the mapping changes.

- Since data is from previous years, it prevents being able to give timely feedback to NIST staff, so that they can adjust their training and mapping practices.
- The lack of relevant details for many CVEs continues to introduce time-consuming analysis and variability in mapping results, combined with increasing preference to analyze references more closely.
- Even within the CWE Top 25 Team itself, different analysts can be inconsistent in which CWE mappings they choose for the same CVE, especially for vulnerabilities that do not have very clear phrasing about the weakness. It is not clear whether this is a limitation of CWE itself, variations in terminology within CVE descriptions, or of the varying perspectives and levels of experience of the analysts who perform the mappings.
- Once the remapping task is complete, the version of NVD that was originally used is typically a few months old - for this year, NVD from March 18, 2021, was used. This can cause apparent inconsistencies for those who want to replicate the metric, since new CVE records continue to be added for earlier years.

Once the remapping task is complete, the version of NVD that was originally used is typically a few months old - for this year, NVD from March 18, 2021, was used. This can cause apparent inconsistencies for those who want to replicate the metric, since new CVE records continue to be added for earlier years.

Emerging Opportunities for Improvement

Despite the current limitations of the remapping task, several activities have taken shape recently that might show positive improvements for NVD/CWE mapping data as used in future Top 25 lists:

- NIST's Collaborative Vulnerability Metadata Acceptance Process (CVMAP) program is gaining traction, with positive interactions with CVE Candidate Numbering Authorities (CNAs) that are likely to improve CWE mapping quality from those CNAs. The CWE Program plans to have closer interaction with CNAs to help obtain more precise data.
- Version 5.0 of the CVE JSON record format includes direct support for including CWE mappings in CVE records, which seems likely to improve the quality and precision of CWE mappings.
- In March 2021, the CWE Program released CVE->CWE Mapping Guidance, which makes it easier for CNAs and other parties to perform the technical task of finding appropriate CWE mappings for their vulnerabilities.

Acknowledgements

The 2021 CWE Top 25 Team includes (in alphabetical order): Adam Chaudry, Steve Christey Coley, Kerry Crouse, Kevin Davis, Devon Ellis, Parker Garrison, Christina Johns, Luke Malinowski, Rushi Purohit, Becky Powell, David Rothenberg, Alec Summers, and Brian Vohaska. Members of the NIST NVD Team that coordinated on the Top 25 include Christopher

Turner, Robert Byers, and Vidya Ananthakrishna. Finally, thanks also to the broader CWE community for suggesting improvements to the process.

Archive

Past versions of the CWE Top 25 are available in the Archive.