CVSS: Ubiquitous and Broken

HENRY HOWLAND, Drew University, USA

The Common Vulnerability Scoring System is at the core of vulnerability management for systems of private corporations to highly classified government networks, allowing organizations to prioritize remediation in descending order of risk. With a lack of justification for its underlying formula, inconsistencies in its specification document, and no correlation to exploited vulnerabilities in the wild, it is unable to provide a meaningful metric for describing a vulnerability's severity, let alone risk. As it stands, this standard compromises the security of America's most sensitive information systems.

CCS Concepts: • **Security and privacy** → **Formal methods and theory of security**; *Security requirements*; • **Networks** → Network reliability;

Additional Key Words and Phrases: Threat and vulnerability management, security standards, SCAP, CVSS, remediation prioritization

ACM Reference format:
Henry Howland. 2023. CVSS: Ubiquitous and Broken. *Digit. Threat.: Res. Pract.* 4, 1, Article 1 (March 2023), 12 pages.
https://doi.org/10.1145/3491263

1 INTRODUCTION

Common Vulnerability Scoring System (CVSS) is a deeply rooted information security standard used to gauge the severity of **Common Vulnerabilities and Exposures (CVEs)** on a scale of 1 (Low) to 10 (Critical). Its first iteration, CVSS v1, was introduced in late 2005 by the National Infrastructure Advisory Council [25, 37]. There were other vulnerability scoring systems in the 2000s, but unlike them, CVSS was sponsored by the **National Institute for Standards and Technology (NIST)** [39]. CVSS v1 was deemed non-viable and quickly replaced with CVSS v2 in 2007, with all the CVE's CVSS scores subsequently backfilled with CVSS v2 scores [44]. CVSS v3.1, the most recent version released in 2019, shares the same rating scheme as CVSS v3, released in 2015, with only a few differences in features and intent [47]. Currently, the standard is maintained by the **Forum of Incident Response Teams (FIRST)** through the **CVSS Special Interest Group (CVSS-SIG)**. Since CVSS v3 and CVSS v3.1 have identical scoring formulas, CVSS v3 will be used to refer to both of them. Additionally, the **Common Platform Enumeration (CPE)** system is used to describe classes of applications, operating systems, and hardware devices. Each CVE is given a list of affected CPEs and a CVSS score. CVSS, CVE, and CPE are standards contained under the umbrella of the Security Content Automation Protocols, a series of tools, languages, metrics, and identification schemes used to help manage and document security content [43].

The **National Vulnerability Database (NVD)** is a complement to Security Content Automation Protocol systems, particularly CVE. Acting as a centrally managed repository of CVEs, the NVD and has been curated by NIST since 2005 [12]. The NVD experienced many growing pains in its first few years, as a vendor agnostic

Author's address: H. Howland, Drew University, P.O. Box 1212, Madison, New Jersey, USA, 07940; email: hhowland@drew.edu.

2576-5337/2023/03-ART1 $15.00
https://doi.org/10.1145/3491263

database for cataloging computer vulnerabilities was a novel idea. While slow at times, the NVD has performed well at its purpose, and is used worldwide as a primary source for computer vulnerabilities. Currently, the NVD contains over 150,000 CVEs and is growing rapidly with the proliferation of software, vulnerability testing capabilities, and the expansion of CVE reporting infrastructure as shown in Figure 1 below [1]. While these standards and systems complement each other as they all originated from NIST, it is important to note that the NVD, CVE, and CPE are currently U.S. initiatives, while CVSS is now an international standard [44].

2 RELATED WORK

This work relies on research related to the efficacy of CVSS, its original purpose, as well as other risk scoring systems. Full context of the ecosystem surrounding CVSS is needed to understand how it has developed and its current impact on the industry.

The search for related works was guided by four questions. Are there any other systems that try to capture the severity or risk imposed by a single vulnerability? Is CVSS being used correctly? How effective is CVSS v3 at measuring the severity/risk? Where is CVSS mandated for use? As for the scope of publication venues, review was not limited to academic publications as there are many valuable insights from information technology companies and the security practitioners using the standard. This is especially the case for security practitioners as their security operations expertise and time spent applying the standard may allow for a more useful perspective on CVSS than an academic researcher could provide. Special effort was also given to researching CVSS documents related to FIRST and NIST due to their involvement in the standard's creation and proliferation.

As this article discusses CVSS v3's role as a measure of risk, an awareness of the different documents defining information system related risk would be useful. For instance, the International Organization for Standardization defines risk as the "effect of uncertainty on objectives" in their *ISO Guide 73* [30]. Various NIST publications such as the *Federal Information Processing Standards 200* discuss risk and risk reduction controls as well [28]. In the context of this article, the definition of risk given in the *NIST SP 800-30 Rev.1 Guide for Conducting Risk Assessments* and the *Committee on National Security Systems Information Assurance Glossary* will be used, which states that "Risk is a measure of the extent to which an entity is threatened by a potential circumstance or event, and is typically a function of: (i) the adverse impacts that would arise if the circumstance or event occurs; and (ii) the likelihood of occurrence" [48, 53]. This definition is the most appropriate to frame the discussion in this article as it is a product of NIST like CVSS and CVE and it is more precise than the ISO 73 definition.

Building on the introduction, a closer look at the vulnerability rating systems that preceded and coexist with CVSS would be helpful for contextualizing it. Keeping in mind that CVSS was not successfully launched until 2007, in 2001, the **Computer Emergency Response Team Coordination Center (CERT/CC)** released the Vulnerability Note Field Descriptions, which outlined a set of metrics for describing a vulnerability's severity [15, 38]. SANS's similar solution came in 2003, and there were various vendor specific severity scores throughout this time in the early 2000's as well [27, 39]. Other more recent and notable scoring systems include DREAD, the SSVC (previously TEMSL), which uses decision trees for scoring, and CVSS adaptations to specific problem sets and industries, such as RSS-MD, IVSS, and RVSS [11, 19, 51, 52, 54].

Papers relating to the **Exploitation Prediction Scoring System (EPSS)** as well as any other studies on exploit predication are also highly relevant. This line of research dates back to 2010 with the article *Beyond Heuristics: Learning to Classify Vulnerabilities and Predict Exploits*, and with more high quality information security data sources available than ever before, research avenues for finding cyber-attack predictors and vulnerability remediation mechanisms have flourished in the past few years [13, 31]. Additionally, any exploit prediction work related to Kenna Security (recently acquired by Cisco) is pertinent to this article as they were responsible for creating the datasets used in several volumes of research demonstrating CVSS's inefficiencies, as well as proposing the EPSS [32, 33]. Interestingly, the EPSS is now supported by FIRST, the maintainers of CVSS [10].

There are myriad articles notable for critiquing CVSS and discussing its performance as a risk score. Much of these critiques are summated in the 2018 whitepaper, *Toward Improving CVSS* [50]. This article was written

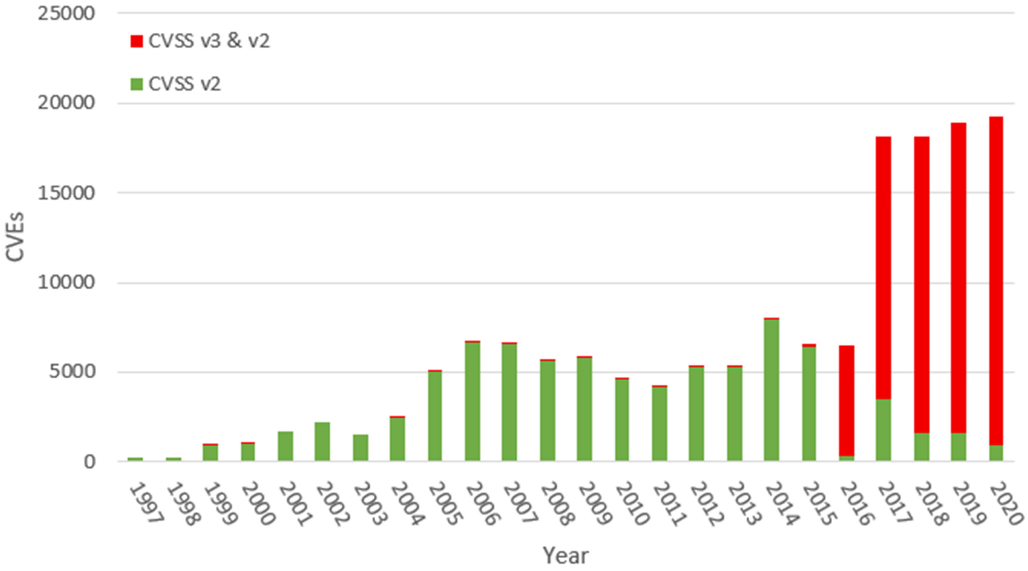

Fig. 1. Yearly published CVEs by CVSS score version.

by several individuals involved in the development of CVSS at the CERT/CC, prompting an updated version of the standard to be released a couple months later. Another notable work, *Comparing Vulnerability Severity and Exploits Using Case-Control Studies*, determined that CVSS's scoring system produced no tangible reduction in risk [5]. In fact, multiple major tech companies have commented on miscellaneous issues with CVSS. For instance, a letter written to the CVSS-SIG from the company Risk-based Security titled *CVSSv2 Shortcomings, Faults, and Failures Formulation* provides an excellent analysis of some of the technical issues that beset CVSS v2 [23]. A RedHat security architect has also written about CVSS's poor risk assessment performance in a blog post titled *Why CVSS does not equal risk: How to think about risk in your environment* [46]. Just as well, the chief architect for Capitol One's container product, Critical Stack, wrote an article on their issues with CVSS [42]. The McAfee Corporation also posted a case study of a false negative where the authors, with decades of combined InfoSec experience, state that CVE-2017-3735 was rated too highly due to structural issues in CVSS [49]. Gunter Ollman, a well-known security executive has felt it necessary to write an article in *Security Weekly* entitled *Stop Using CVSS to Score Risk* to articulate his concerns with CVSS [41].

To better understand how CVSS has become ubiquitous it would be useful to have a familiarity with the articles that describe and mandate its use across the government and industry. One of the most widely cited examples of CVSS being mandated is the ***Payment Card Industry (PCI)*** *Data Security Standard* that forces members of the PCI to use CVSS for patch management [45]. Although, it is most likely the many government documents mandating its use that have helped stimulate its spread. Examples of this include the *Department of Defense (DoD) Joint Special Access Program (SAP) Implementation Guide (JSIG)* for securing top-secret programs, the *Binding Operational Directive 19-02*, which specifies vulnerability remediation policies for internet accessible systems, the *DoD Instruction 8531.01 DoD Vulnerability Management*, which establishes and renews CVSS-based vulnerability management practices accross the DoD Information Network, and last, the *FedRAMP Vulnerability Scanning Requirements* created the vulnerability management practices for cloud-based federal systems [2, 3, 9, 20].

In general, this article aims to contribute to the field of vulnerability management by demonstrating that CVSS is not a misunderstood and misused system created to rate the severity of a vulnerability, but was intended from the beginning and has always been a measure of vulnerability risk, thereby changing the narrative surrounding the standard and allowing it to be more properly evaluated [50]. This article builds on a body of CVSS v2 efficacy research, coming to similar conclusions for CVSS v3 using different methods. This work also fills a literature gap on CVSS v3 efficacy testing as most of the CVSS efficacy research, including the literature published on EPSS, utilized CVSS v2 data [5, 23, 32, 33]. There are also fewer articles analyzing the source documentation and historical context of CVSS as is done here. Lastly, this article contributes to the conversation by attempting to convey CVSS's far reaching impact on national security.

3 A RISK SCORE WITHOUT ACCOUNTABILITY

CVSS has consistently been used as a risk score. It requires little effort and is inexpensive to say that a CVE with a CVSS score of 9 needs to have priority patching over one with a 7 without consideration for the context surrounding the vulnerability. The CVSS-SIG recognized this issue and after many complaints, particularly those outlined in *Toward Improving CVSS*, developed and released the specification documents for CVSS v3.1 in June of 2019 [1, 50]. The title of Section 2.1 in the CVSS v3.1 User Guide plainly states that "CVSS Measures Severity, not Risk" [26]. While the CVSS-SIG never goes as far to say what the distinction is, this move, intended to separate CVSS from its perception as a risk score, contradicts the documents originally used to launch the standard. For example, CVSS v2, the first widely deployed version, was launched in June 2007. Two months later, the lead NIST architects of CVSS v2 released the document *The Common Vulnerability Scoring System and Its Applicability to Federal Agency Systems*. The introduction thereof states that "IT management must identify and assess vulnerabilities across many disparate hardware and software platforms. They (IT management) need to prioritize these vulnerabilities and remediate those that pose the greatest risk. The **Common Vulnerability Scoring System (CVSS)** is an open framework that addresses this issue" [38]. The authors of CVSS make it clear here, and in the rest of this document, that the primary intention behind CVSS is IT risk management. As such, CVSS has been mandated for use by the Payment Card Industry, Department of Homeland Security, and the DoD to treat risk as well as being recommended by NIST for the same purpose [2–4, 45, 48]. Stating that CVSS is not a risk score does not change the fact that this is its original purpose and primary application.

The CVSS-SIG's attempt to separate CVSS from its perception as a risk score is also contradicted by CVSS v3.1's own specification document. Under Section 3.1 "CVSS Measures Severity Not Risk" the authors state that a real risk assessment system considers "factors outside the scope of CVSS such as exposure and threat" even though CVSS has facilities for measuring both [26]. The intention behind the Environmental Score of CVSS was to allow users to modify the CVSS score depending on the environment a vulnerability is found in, specifically, characteristics of a vulnerability's environment "that affect Exploitability, Scope, or Impact can be reflected via an appropriately modified Environmental Score" [37]. The Environmental Score's basis for modifying a vulnerability's severity based on Exploitability and Scope clearly align with the exposure risk assessment factor described above. Furthermore, measurements for threat can be found in the **Exploit Code Maturity (ECM)** metric, a member of CVSS v3's Temporal Metric Group [37]. Potential values of the ECM include Not Defined, High, Functional, Proof of Concept and Unproven. Naturally, the ECM of a CVE relates directly to the threat of exploitation of a vulnerability, as the measure of exploitability is a measure of threat [37]. If the ECM is High, then this vulnerability is much more likely to be successfully exploited than if the ECM is just functional. Denying this is analogous to comparing a soldier with a musket to one with a modern rifle and judging them to be equal threats. With Exploit Code Maturity, CVSS is able to take in vectors for threat and vulnerability into its formula, the two fundamental factors for modeling cyber risk as conveyed by NIST and the Committee on National Security Systems [48, 53]. Exploit Code Maturity is an inappropriate metric if CVSS is meant to be solely a vulnerability score. The effort to change the standard from being perceived as a risk score in the CVSS v3.1 specification document is a clear attempt to shirk CVSS's responsibility as a means of measuring risk, not because it is not a risk score, because it performs poorly.

This issue is further exacerbated by FIRST's stated intentions for CVSS v4 in their presentation, *The State of CVSS for the 2020s*. At the top of the list of improvements for CVSS is "Threat Intelligence Metrics" with the example given of "Likelihood of Attack" [22]. There is no reason to have this metric, or any other threat intelligence metric in the standard if it is a technical severity score. The only value that threat intelligence metrics have for a CVE is capturing the risk a CVE introduces in an environment.

4 CONTEXTUAL ISSUES OF VULNERABILITY RATING

The issues regarding CVSS's inability to account for a vulnerability's context are well known and widely documented in academic papers, forums, and blogs [35, 41, 42, 46, 49, 50]. As mentioned above, the Environmental Score is the component of CVSS v3 meant to grapple with this issue, although it lacks the granularity required to be effective. The Environmental Score allows for score creation based on the confidentiality, integrity, and availability requirements of a system's information. The metrics and values for each of these scores can be found in Table 1 [37]. Changing the value of any of these variables allows you to move the score up or down one increment, which is simply not enough for a large and diverse environment. Take a typical pharmaceutical company; the availability value for a system managing assembly line manufacturing would be the same as a web server, high. While they both require high availability, the requirement for a manufacturing system would be much higher than that of the web server in a business value comparison. Even if the Environmental Score is used, it would have to be supplemented with a compensatory control, such as GxP. Although this is just one example, any number of scenarios can be suggested where a three value system is not helpful for this application in a modern enterprise environment. A method that explicitly takes into account the value lost from a computer unable to provide its service for the duration of its mean time to recovery may be more useful.

Table 1. Environmental Score Valaues

Metric Value	Numerical Values
Not Defined (X)	1
High (H)	1.5
Medium (M)	1
Low (L)	0.5

Quantification of the Confidentiality Impact in the Impact Metrics is also a weakness in CVSS's rating scheme. The Confidentiality Impact is defined roughly as the amount of data that a vulnerability gives you access to in its context, but this approach is incapable of accounting for the value of the data that may be leaked due to a vulnerability. Take CVE-2014-3566, aka POODLE; it is a man in the middle vulnerability with a CVSS v3 score of 3.4, which allows the attacker to decrypt a small amount of ciphertext traveling over SSL 3.0. The revealed information from this vulnerability could be benign or sensitive personally identifiable information, either way, CVSS is unable to account for it with the Confidentiality Impact metric. An argument could be made that this is where the Environmental Score is meant to be applied, but it would not be useful for three reasons. One, when POODLE was disclosed, SSL 3.0 was utilized by nearly every device connected to the internet, it would be impractical to apply the Environmental Score to all of them in an organization [40]. Two, it is often difficult to predict what kind of data is being transported over a particular system, which is necessary to determine the Confidentiality Requirement of the Environmental Score. Three, the vulnerability is already so low that, even if the Confidentiality Requirement is set to High, the CVSS v3 Environmental Score still only comes out to 4.2, just barely a medium severity vulnerability according to CVSS's severity levels shown in Table 2 [24, 37]. Despite the fact that POODLE is not severe according to CVSS, it is one the highest profile vulnerabilities of the recent decade. With, at the time, over one million public IP addresses vulnerable to the attack, the government recognized its impact quickly and not three days after POODLE's original disclosure did the Cybersecurity and Infrastructure Security Agency issue a bulletin on the vulnerability urging people to disable SSL 3.0 [18, 29]. Examples like this where elements of the greater context of a vulnerability cannot be taken into account make CVSS prone to false negatives and positives.

Table 2. CVSS Severity Rating

Metric Value	Severity Level
0–4	Low
4–7	Medium
7–9	High
9–10	Critical

5 CVSS SEVERITY MEASUREMENT EFFICACY

Vulnerabilities that are more severe should in principle mean that they are more attractive candidates to develop exploitation and weaponization mechanisms for. As severity has a direct relationship to the value that exploitation of a vulnerability provides an attacker, a more severe vulnerability would in principle allow a greater impact on a given environment [21]. This in mind, if CVSS is an accurate gauge of a vulnerability's severity, there should be a linear correlation between the CVSS v3 score of a CVE and the likelihood of that CVE having been presently or historically weaponized. For the purposes of this article, a weaponized vulnerability refers to a vulnerability that has been exploited in the wild.

CVSS's predictive capability for weaponization will be tested using a sample of CVEs with CVSS v3 scores where the weaponization status can be reliably determined. A CVE's weaponization status is decided using information available in the **Qualys Knowledge Base (QKB)** associated with each CVE. Qualys Inc., a company focused on enterprise vulnerability management solutions, maintains exploitation information on the CVEs it tracks in the QKB with a team of analysts. Of the 152,159 CVEs captured in the NVD circa late 2020, 70,140 (or 54%) are assigned CVSS v3 scores. Of these 70,140 CVEs, 28,779 (or approximately 41%) of CVEs are tracked by the QKB. All of the data was downloaded and processed with Python, SPSS, and other data analysis tools.

To test for similarity of affected platforms between the NVD and the QKB subset, the CPE tags of each vulnerability were parsed out of the NVD's CVE metadata. These tags contain the name of the affected product and its developer, and as one vulnerability can affect multiple of a developer's products, or versions of those products, there can be multiple CPE tags per CVE. For instance, CVE-2017-5754 is a hardware vulnerability known as Meltdown, which affected 208 Intel products. Due to the large number of distinct product and developer combinations in the QKB, 12,776 in total, only ratios based on the distinct developers, of which there are 1,524, will be shown here. A ratio comparison for the top 10 developers between the QKB and NVD is shown below in Table 3.

Table 3. Publishing Body Ratios

NO	QKB Developer	QKB Percent	QKB Count	NVD Developer	NVD Percent	NVD Count
1	Microsoft	14.154%	6,245	Microsoft	8.268%	7,467
2	Google	8.559%	3,782	Google	5.431%	4,905
3	Apple	7.982%	3,527	Apple	4.448%	4,017
4	Debian	7.274%	3,214	Oracle	4.235%	3,825
5	Adobe	4.524%	1,999	Debian	3.742%	3,380
6	Oracle	4.171%	1,843	IBM	3.113%	2,011
7	Ubuntu	5.094%	1,809	Cisco	2.668%	2,049
8	Red Hat	3.886%	1,717	Adobe	2.423%	2,168
9	Linux (kernel)	3.784%	1,672	Red Hat	2.34337%	2,119
10	openSUSE	2.591%	1,145	Linux (kernel)	2.239%	2,022

It would appear that Qualys may favor capturing Microsoft and big-tech-based vulnerabilities but this does not take into account outliers. Despite having over 40% of the known vulnerabilities in the NVD, the QKB again has only 1,524 distinct software developers compared to the NVD's 8,946. The NVD has a very long tail of software developers with one or two vulnerabilities, skewing the data such that all the NVD ratios are smaller. The calculations are done again in Table 4 taking the top 500 developers by their vulnerabilities published from the two datasets to partially correct for the NVD's larger pool of developers.

Table 4. Adjusted Publishing Body Ratios

NO	QKB Developer	QKB Percent	QKB Count	NVD Developer	NVD Percent	NVD Count
1	Microsoft	14.597%	6,245	Microsoft	10.390%	7,467
2	Google	8.827%	3,782	Google	6.825%	4,905
3	Apple	8.231%	3,527	Apple	5.589%	4,017
4	Debian	7.501%	3,214	Oracle	5.323%	3,825
5	Adobe	4.666%	1,999	Debian	4.703%	3,380
6	Oracle	4.302%	1,843	IBM	3.911%	2,011
7	Ubuntu	4.222%	1,809	Cisco	3.352%	2,049
8	Red Hat	4.007%	1,717	Adobe	3.044%	2,168
9	Linux (kernel)	3.902%	1,672	Red Hat	2.946%	2,119
10	openSUSE	2.672%	1,145	Linux (kernel)	2.813%	2,022

The developer ratios adjusted for outliers are much more similar, although the QKB still has a higher percentage of Microsoft-based vulnerabilities as well as some other large tech companies. This is possibly because Qualys has developed vulnerability scanning and tracking capabilities tailored to an enterprise environment.

Checking for CVSS score representativeness is much more straight forward, and is done by comparing the descriptive statistics of the two datasets. These values are captured in Table 5 below and show that the CVSS scores of the QKB are representative of those found in the NVD.

Table 5. CVSS Data Set Statistics

CVSS v3 Data Set	Avg CVSS v3	Median CVSS v3	STD CVSS v3	Total CVEs
Full NVD	7.254	7.5	1.657	70,140
QKB NVD Sample	7.290	7.5	1.607	28,779 (41%)

Overall, while these findings indicate a small bias in the QKB's types of vulnerabilities favoring large tech companies, the QKB is still useful for analysis due to how many vulnerabilities from the NVD it includes, and the fact that its CVSS score distribution is so similar. Even if the results from the analysis on the QKB are not generalized to the rest of the NVD, the results are still representative of a large swath of the NVD. In addition, any analysis done on this dataset would presumably be relevant to an enterprise IT environment as that is who Qualys maintains the QKB for.

Moving on to weaponization analysis, the CVE weaponization status is not explicitly defined in the QKB, so it has been derived from the Threat Intel Values (TIV)s that are associated with each CVE within the QKB. A CVE is confirmed to have been weaponized if the TIV, Exploit Kit, Malware, or Active Attacks, is associated with it. While there are TIVs that indicate the development of PoC exploit code, this does not necessarily mean that the code was used by a threat actor. The distribution of weaponization values across floored CVSS v3 values in the sample data set is shown in Figure 2 and Table 6.

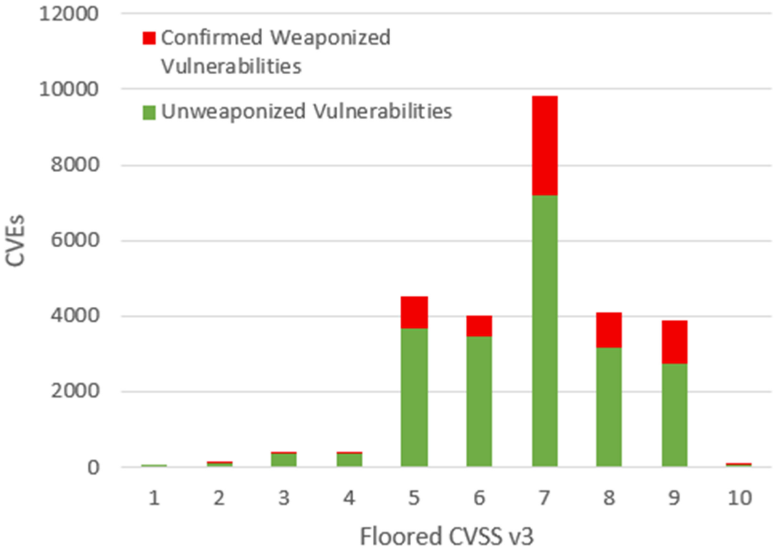

Fig. 2. Weaponized vs Unweaponized CVEs by CVSS score.

Table 6. Weaponized Versus Unweaponized CVEs by CVSS Score

Floored CVSS v3	Not Weaponized	Weaponized	Total	Percentage of Total Weaponized
1	4	0	4	0%
2	98	15	113	13.3%
3	360	60	420	14.3%
4	360	60	420	14.3%
5	3653	862	4,515	19.1%
6	3454	564	4,018	14.0%
7	7208	2,629	9,837	26.7%
8	3147	927	4,074	22.8%
9	2723	1,149	3,872	22.8%
10	55	15	70	21.4%
Total	22060	6,547	28,607	

As the data in Figure 2 and Table 6 indicate, there is no linear relationship between a CVE's CVSS v3 score and its weaponization status. In fact, it may be the case that CVEs with floored CVSS v3 scores of 7 are actually the most severe on average, measuring severity by their likelihood of actual exploitation. We can also see an issue of poor overall spread of the CVSS scores, which is shared by the weaponization of the CVEs. If a resource-constrained IT organization set out to minimize risk by increasing their resistance to attacks, then they would be better off patching any random 100 CVE's with a CVSS score of 7 than they would by doing the same for a group of CVSS 10s. These findings for CVSS v3 fall in line with studies of CVSS v2, which similarly found that remediating all vulnerabilities with a high severity was largely ineffective at stopping cyber-attacks [5, 31]. These results indicate that CVSS has thus far provided little IT risk management value other than giving regulatory bodies a mechanism to enforce system patching.

6 THE WRONG FACTORS—SCORING ISSUES

One possibility as to why there is no linear correlation between CVE weaponization and the CVSS score could be a result of the comprising vectors being weighted incorrectly. The question then becomes, could CVSS be made more useful if a factor such as Access Complexity or Attack Vector were weighted more heavily? As each vector is roughly ordinal in its weighting, as shown in Table 2 and the CVSS specification document, a correlation between the CVSS scoring vectors and weaponization can be tested as shown below in Table 5 [37].

Table 7. Paired T-test of Most Common TF Families for Pearson Correlations

One Tailed Correlation	Factor 1	Factor 2	Rho	Correlation Strength
Pearson	CVSS v3 Base (ref)	Weaponization	0.099	None
Spearman	Access Complexity	Weaponization	−0.031	None
Spearman	Attack Complexity	Weaponization	−0.081	None
Spearman	Attack Vector	Weaponization	−0.081	None
Spearman	Availability	Weaponization	0.047	None
Spearman	Confidentiality	Weaponization	0.033	None
Spearman	Integrity	Weaponization	0.135	Very Weak
Spearman	Privileges Required	Weaponization	0.180	Very Weak
Spearman	Scope	Weaponization	−0.089	None
Spearman	User Interface	Weaponization	−0.049	None

As Table 7 shows, there is no one vector in CVSS that can be correlated to the weaponization of a vulnerability. While these vectors are great technical descriptors of the vulnerability itself and have many uses outside of producing a CVSS score, there does not appear to be a way to modify the weightings of the CVSS v3 metrics that would allow for more efficient vulnerability remediation and risk reduction.

7 DISCUSSION ON ADOPTION AND IMPACT

CVSS was the first attempt by the federal government at developing a vulnerability remediation prioritization system. There are no indications of efficacy testing ever having taken place, and the standard itself is poorly maintained. As of Q2 2021, CVSS's scoring system is only on its second version in its 14-year history as the first version was scrapped immediately after creation with CVSS v2 released in June 2007, effectively acting as the first deployed version. The scoring system did not go under major revisions until early 2015 with the development of CVSS v3.0 ending in June. The CVSS-SIG's slow time to react to the problems of CVSS has allowed for its deep inefficiencies to become engrained in the security industry and America's most sensitive institutions.

CVSS is used as a core specification for creating and maintaining classified and secure information systems across the government. FedRAMP is a framework for securely building and transferring government applications to the cloud, and its *FedRAMP Vulnerability Scanning Requirements* state that "For any vulnerability with a CVSS v3 base score assigned in the latest version of the NVD, the CVSS v3 base score must be used as the original risk rating" [19]. To pass and maintain **Authorization to Operate (ATO)** under FedRAMP, entities must remediate vulnerabilities according to CVSS. Additionally, the DoD's procedure for vulnerability management, *DoD Instruction 8531.01 DoD Vulnerability Management* uses CVSS as the primary metric for determining vulnerability risk, stating that "CVSS provides a uniform and standardized vulnerability scoring method, an open framework, and an ability to prioritize risk" [20]. The use of CVSS for vulnerability management is standard practice across all levels of information security requirements, even extending up to special access programs, the most sensitive programs in the DoD, as relayed by the *Department of Defense (DoD) Joint Special Access Program (SAP) Implementation Guide (JSIG)* [9]. The sentiment that CVSS should be used as a primary risk metric is a result of the original NIST report, *The Common Vulnerability Scoring System (CVSS) and Its Applicability to Federal Agency Systems*, as many governmental documents working with CVSS cite it directly [38]. The national security implications of building and maintaining information security on an untested and dysfunctional risk management system are worrisome.

8 ALTERNATIVES MOVING FORWARD

For years CVSS has been the security vendor's de facto risk score, but to provide more efficient vulnerability management, many vendors have integrated weaponization prediction systems into their products. This is a growing trend that can be seen in offerings from Qualys, Symantec, and Tenable [6, 8, 14]. Some of these systems outright replace CVSS and some work like Tenable's **Vulnerability Prioritization System (VPR)**, where an underweighted CVSS score is used in combination with an exploit prediction score derived from a ML model. While these scoring systems may be more effective at risk reduction than CVSS alone, they are opaque, with very little information provided on their construction, and due to the fact that none of them are open source, they cannot be used by the government to supplement or replace CVSS.

An open source scoring system is needed and at this time there are not many compelling options. Although EPSS seems to be a very promising compliment, as well as the possible basis for a future replacement. Unlike some commercial machine learning models, EPSS uses a model built on publicly available data to predict which CVEs are the most likely to be weaponized. It has shown to be far more efficient than CVSS at patching vulnerabilities that are actually being exploited [33]. While EPSS adds great threat awareness to remediation prioritization efforts, it lacks the ability to account for an individual vulnerability's context in its calculation, and thus is currently unsuitable as a primary risk score. Perhaps information such as vulnerability exposure and device business value could be combined with the threat score of EPSS to create a more comprehensive risk scoring system.

One possible avenue for creating a replacement vulnerability risk scoring system would be a NIST or NSA sponsored contest. This is how NIST chose the Advanced Encryption Standard and is how the military, specifically the Air Force, derives its most advanced and versatile technologies [7, 16, 17, 34]. Considering the sizeable amount of new cyber security companies and the aggressive investment in the information security industry, there would likely be no shortage of high quality entries.

9 CONCLUSION

CVSS is laden with issues. There is no clear reasoning given as to how the system was devised, it is riddled with logical inconsistencies, and it is only able to partially account for the context of a vulnerability, as well as being an empirically poor means of representing a vulnerability's severity. It is unreasonable for NIST or any other organization to recommend, let alone mandate, that CVSS in its current state be used for the purposes of vulnerability management. As CVSS is an artifact of an emerging discipline, it has become intertwined with vulnerability management in all facets of government and industry. As these institutions are legally incapable of stopping their use of CVSS, the CVSS-SIG must make radical changes to the standard in the imminent CVSS v4 backed up by transparent efficacy testing, or a new remediation prioritization system should be adopted [36].

ACKNOWLEDGMENTS

Thanks to Qualys for providing API access to the Qualys Knowledge Base as well as giving background information on how vulnerabilities in its database are maintained.

REFERENCES

[1] 2005. The National Vulnerability Database. Retrieved from https://nvd.nist.gov/.

[2] Brian Conrad. 2018. FedRAMP vulnerability scanning requirements. Retrieved from https://www.fedramp.gov/assets/resources/documents/CSP_Vulnerability_Scanning_Requirements.pdf.

[3] Chris Krebs. 2019. Binding operational directive 19-02. Retrieved from https://cyber.dhs.gov/bod/19-02/.

[4] Murugiah Souppaya and Karen Scarfone. 2013. NIST special publication 800-40 revision 3: Guide to enterprise patch management. Retrieved from https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-40r3.pdf.

[5] Luca Allodi and Fabio Massacci. 2014. Comparing vulnerability severity and exploits using case-control studies. *ACM Trans. Info. Syst. Secur.* 17, 1 (Aug. 2014), 1–20. https://doi.org/10.1145/2630069

[6] Yair Amit. 2019. Symantec Mobile threat defense: Spotlight on modern endpoint vulnerability management. Retrieved from https://symantec-enterprise-blogs.security.com/blogs/product-insights/symantec-mobile-threat-defense-spotlight-modern-endpoint-vulnerability-management.

[7] David C. Aronstein, Michael J. Hirschberg, and Albert C. Piccirillo. 2000. *Advanced Tactical Fighter to F-22 Raptor: Origins of the 21st Century Air Dominance Fighter.* American Institute of Aeronautics and Astronautics.

[8] Cayla Baker-Ruiz. 2019. Predictive prioritization. Retrieved from https://web.archive.org/web/20191017173028/https:/www.tenable.com/predictive-prioritization.

[9] David Beèn and Kenneth Brown. 2016. Department of Defense (DOD) Joint Special Access Program (SAP) Implementation Guide (JSIG). Retrieved from https://www.dcsa.mil/portals/91/documents/ctp/nao/JSIG_2016April11_Final_(53Rev4).pdf.

[10] Robyn Blum, Marty Palka, and Ashley Vandiver. 2021. Cisco announces intent to acquire kenna security to deliver industry leading vulnerability management. Retrieved from https://newsroom.cisco.com/press-release-content?type=webcontent&articleId=2160133.

[11] Clint Bodungen. 2019. IVSS. Retrieved from https://web.archive.org/web/20190327162328/https:/securingics.com/IVSS/IVSS.html.

[12] Harold Booth, Doug Rike, and Greg Witte. 2013. ITL bulletin for December 2013 the national vulnerability database (NVD): Overview. Retrieved from https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=915172.

[13] Mehran Bozorgi, Lawrence Saul, Stefan Savage, and Geoffrey Voelker. 2010. Beyond heuristics: Learning to classify vulnerabilities and predict exploits. Retrieved from http://cseweb.ucsd.edu/~saul/papers/kdd10_exploit.pdf.

[14] Tami Casey. 2019. Qualys brings its market leading vulnerability management solution to the next level. Retrieved from https://www.qualys.com/company/newsroom/news-releases/usa/qualys-brings-its-market-leading-vulnerability-management-solution-to-the-next-level-introducing-vmdr/.

[15] CERTCC. 2001. CERT/CC vulnerability note field descriptions. Retrieved from https://web.archive.org/web/20010816053036/http:/www.kb.cert.org:80/vuls/html/fieldhelp.

[16] Lily Chen. 2016. AES development. Retrieved from https://csrc.nist.gov/projects/cryptographic-standards-and-guidelines/archived-crypto-projects/aes-development.

[17] James Ciborski. 2002. The F-15 eagle: A chronology. Retrieved from https://web.archive.org/web/20030421185317/http://www.ascho.wpafb.af.mil/AFtopics/f15chronology.htm.

[18] CISA. 2014. Alert (TA14-290A). Retrieved from https://us-cert.cisa.gov/ncas/alerts/TA14-290A.

[19] Dawid Czagan. 2014. Qualitative risk analysis with the DREAD model. Retrieved from https://resources.infosecinstitute.com/topic/qualitative-risk-analysis-dread-model/.

[20] Dana Deasy. 2020. DoD instruction 8531.01 DoD vulnerability management. Retrieved from https://www.esd.whs.mil/Portals/54/Documents/DD/issuances/dodi/853101p.pdf?ver=2020-09-15-143058-347.

[21] Debabrata Dey, Atanu Lahiri, and Guoying Zhang. 2015. Optimal policies for security patch management. *INFORMS J. Comput.* 27, 3 (2015), 462–477. https://doi.org/10.1287/ijoc.2014.0638

[22] Dave Dugal and Dale Rich. 2020. The state of CVSS for the 2020s. Retrieved from https://www.first.org/resources/papers/sig-jun2020/CVSS-v4-FIRST-SIG-Update-2020-v3.pptx.

[23] Carsten Eiram and Brian Martin. 2013. The CVSSv2 shortcomings, faults, and failures formulation. Retrieved from https://www.riskbasedsecurity.com/reports/CVSS-ShortcomingsFaultsandFailures.pdf.

[24] FIRST. 2015. Common vulnerability scoring system version 3.0 calculator. Retrieved from http://www.first.org/cvss/calculator/3.0.

[25] FIRST. 2015. Introduction to CVSS. Retrieved from www.first.org/cvss/v1/intro.

[26] FIRST. 2019. CVSS v3.1 user guide. Retrieved from https://www.first.org/cvss/user-guide.

[27] Jeff Forristal. 2003. SANS critical vulnerability analysis archive. Retrieved from https://web.archive.org/web/20031202031252/http:/www.sans.org:80/newsletters/cva/.

[28] Carlos Gutierrez and William Jeffrey. 2006. Federal information processing standards 200. Retrieved from https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.200.pdf.

[29] Imarc. 2014. The POODLE vulnerability: Is the dog's bark worse than the bite? *Security Scorecard.* Retrieved from https://securityscorecard.com/blog/poodle.

[30] ISO. 2009. ISO guide 73:2009: Risk management—Vocabulary. Retrieved from https://www.iso.org/standard/44651.html.

[31] Jay Jacobs, Sasha Romanosky, Idris Adejrid, and Wade Baker. 2019. Improving vulnerability remediation through better exploit prediction. Retrieved from https://arxiv.org/abs/1908.04856.

[32] Jay Jacobs, Sasha Romanosky, Idris Adjerid, and Wade Baker. 2020. Improving vulnerability remediation through better exploit prediction. *J. Cybersecur.* 6, 1 (Sep. 2020). https://doi.org/10.1093/cybsec/tyaa015

[33] Jay Jacobs, Sasha Romanosky, Ben Edwards, Idris Adjerid, and Michael Roytman. 2020. Exploit prediction scoring system (EPSS). Retrieved from https://www.first.org/epss/.

[34] Gerard Keijsper. 2007. *Joint Strike Fighter—Design and Development of the International Aircraft*. Pen & Sword Books Ltd.

[35] Dan J. Klinedinst. 2015. CVSS and The Internet of Things. *CVSS and the Internet of Things*. Retrieved from https://insights.sei.cmu.edu/cert/2015/09/cvss-and-the-internet-of-things.html.

[36] Art Manion and Jake Koun. 2020. Vulnerability prioritization and disclosure—The right security. *Risk-based Security*. Retrieved from http://www.riskbasedsecurity.com/2020/12/15/vulnerability-prioritization-and-disclosure-the-right-security/.

[37] Adam Maris et al. 2015. CVSS v3.1 specification document. Retrieved from http://www.first.org/cvss/v3.1/specification-document.

[38] Peter Mell, Karen Scarfone, and Sasha Romanosky. 2007. The common vulnerability scoring system (CVSS) and its applicability to federal agency systems. Retrieved from https://www.govinfo.gov/content/pkg/GOVPUB-C13-19c8184048f013016412405161920394/pdf/GOVPUB-C13-19c8184048f013016412405161920394.pdf.

[39] Microsoft. 2001. Microsoft exploitability index. Retrieved from http://www.microsoft.com/en-us/msrc/exploitability-index.

[40] Bodo Möller, Thai Duong, and Krysztof Kotowicz. 2014. This POODLE bites: Exploiting the SSL 3.0 fallback. Retrieved from https://www.openssl.org/~bodo/ssl-poodle.pdf.

[41] Gunter Ollman. 2019. Stop using CVSS to score risk. *Stop Using CVSS to Score Risk*. Retrieved from https://www.securityweek.com/stop-using-cvss-score-risk.

[42] Yenifer Prajapati. 2018. A review of the common vulnerability scoring system. Retrieved from https://medium.com/critical-stack/a-review-of-the-common-vulnerability-scoring-system-2c7d266eda28.

[43] Stephen Quinn, David Waltermine, Christopher Johnson, Karen Scarfone, and John Banghart. 2009. The technical specification for the security content automation protocol (SCAP): SCAP version 1.0. Retrieved from https://csrc.nist.gov/publications/detail/sp/800-126/archive/2009-11-05.

[44] Gavin Reid, Peter Mell, and Karen Scarfone. 2007. CVSS v2 history. Retrieved from http://www.first.org/cvss/v2/history.

[45] Ellen Richey et al. 2018. Payment card industry (PCI) data security standard. https://www.pcisecuritystandards.org/document_library?category=pcidss&document=dss4aag.

[46] Christopher Robinson. 2019. Why CVSS does not equal risk: How to think about risk in your environment. Retrieved from https://www.redhat.com/en/blog/why-cvss-does-not-equal-risk-how-think-about-risk-your-environment.

[47] Harry Saunders. 2015. FIRST announces availability of new common vulnerability scoring system (CVSS) release. Retrieved from http://www.first.org/newsroom/releases/20150610.

[48] Richard Schaeffer. 2010. CNSS instruction no. 4009: National information assurance (IA) glossary. (2010). Retrieved from https://web.archive.org/web/20120227163121/. http:/www.cnss.gov/Assets/pdf/cnssi_4009.pdf.

[49] Brook Schoenfield and Damian Quiroga. 2018. Don't substitute CVSS for risk: Scoring system inflates importance of CVE-2017-3735. Retrieved from https://www.mcafee.com/blogs/other-blogs/mcafee-labs/dont-substitute-cvss-for-risk-scoring-system-inflates-importance-of-cve-2017-3735/.

[50] Jonathan Spring, Eric Hatleback, Allen Householder, Art Manion, and Deanna Schick. 2018. Towards improving CVSS. https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=538368.

[51] Jonathan M. Spring, Eric Hatleback, Allen Householder, Art Manion, and Deanna Shick. 2020. Proritizing vulnerability response: A stakeholder-specific vulnerability categorization (version 1.1). Retrieved from https://weis2020.econinfosec.org/wp-content/uploads/sites/8/2020/06/weis20-final6.pdf.

[52] Ian Stine. 2017. *A Cyber Risk Scoring System for Medical Devices*. Ph.D. Dissertation. Department of the Air Force. Retrieved from https://apps.dtic.mil/sti/pdfs/AD1054765.pdf.

[53] Gary Stoneburner, Alice Gougen, and Alexis Feringa. 2012. NIST special publication 800-30 guide for conducting risk assessments. Retrieved from https://csrc.nist.gov/publications/detail/sp/800-30/rev-1/final.

[54] Víctor Mayoral Vilches, Endika Gil-Uriarte, Irati Zamalloa Ugarte, Gorka Olalde Mendia, and Rodrigo Izquierdo Pisón. 2019. Towards an open standard for assessing the severity of robot security vulnerabilities, the Robot Vulnerability Scoring System (RVSS). Retrieved from https://arxiv.org/abs/1807.10357.

Received 25 March 2021; revised 1 September 2021; accepted 9 September 2021