Understanding the Vulnerability Scoring System through Comparative Analysis

[1]Gagandeep Chawla, [2]Dr. Neeraj Sharma, [3] Dr. Narender Kumar Rawal
[1]Research Scholar, [2]Dean & Professor [3]Assistant Professor
[1]Computer Science,
[1]I.K. Gujral Punjab Technical University, Kapurthala, India.

Abstract: A successful cyber-attack can make a loss of confidentiality, availability and integrity of the organization. Almost all software are having vulnerabilities of one form or another. Increasing demand and use of software almost in every field invites attackers to crack the system and perform malicious activities. Some vulnerabilities causes system to crash and others may cause loss of connectivity. Vulnerabilities in software's such as Buffer Overflow, Race condition and Invalidated Input are the major source from an intruder that can enter in the system. To overcome these issues IT companies and developers are using different vulnerability scoring systems. These scoring systems are used to produce numerical scores of vulnerabilities reflecting their severity. One of the most famous scoring systems used by developers is CVSS (Common Vulnerability Scoring System). It is an open industry standard for rating IT vulnerabilities and helping developers to prioritize vulnerabilities. The purpose to use these scoring systems is to find out the fundamental characteristics of vulnerabilities so that issue can be resolved on severity basis. Most of the vulnerabilities found after use by millions of users. Even some vulnerabilities are never reported and these vulnerabilities are called zero-day vulnerabilities. In this paper, we presented a discussion on the three vulnerability scoring systems i.e. (CVSS, VRSS and CWSS) which will help developers to understand the scoring system so as to prioritize vulnerabilities.

Index Terms: Vulnerability, CVSS, VRSS, CWSS, Threats

I. INTRODUCTION

It is widely accepted that the ignorance of security measures while developing hardware and software's are offering intruders to attack and steal important information. This could be a serious issue which can cause financial loss to the company. Vulnerability is an error, weakness or a glitch in the software or an application, which when exploited results in a negative impact on confidentiality [1]. Intruders can enter the system through these loopholes and can perform malicious activities. Negligence of security measures and pressure to complete the projects on time are the major causes that can increase the presence of vulnerabilities. Security team of IT companies and individual developers always look for systems which can provide them clear image of vulnerabilities. Before using any scoring system a developer should understand the risk associated with the concerned software. A small security mistake can lead to a loss of important business data or confidential information. Threats can come from any directions, so having an adequate vulnerability scoring system can reduce the risk of threats. IT developers should be very conscious while choosing scoring system in order to secure their software against all vicious and dangerous attacks [2]. It is also important to use patches to fix the vulnerabilities and be safe against

system hack. In general practice many users and developers rarely use the patches in timely manner hence increasing the risk of attack.

Number of schemes available for prioritizing vulnerabilities to determine the associated risk level. CVSS is the most accepted, famous and widely used system which lets the developer to find out scores of vulnerabilities. Other scoring systems like Vulnerability Rating and Scoring System (VRSS) and Common Weakness Scoring System (CWSS) are also available that prioritize the vulnerabilities [3,4]. It is assumed that in real world the security services such as access control, authentication of cryptosystem are broken all the time [5].

II. REVIEW OF LITERATURE

In order to analyze and compare different scoring systems, an intensive literature survey of CVSS and other available Vulnerability Scoring systems is conducted. As per this review of literatures we found that almost every scoring system using its own method to score vulnerabilities.

Wang and Gao [2] in his paper "An improved CVSS-based vulnerability scoring mechanism" provides the more accurate score by simplifying the process of evaluating. They have added one more factor "Host Environment" in original base score equation to calculate the vulnerability score. They considered "HE" as an important factor to be taken in base score equation because every operating environment has different impact and ability to handle vulnerabilities. For e.g. Windows and other operating systems are more prone to vulnerabilities rather than Linux/Unix.

Liu and Zhang [11] has proposed a new scoring methodology called VRSS "Vulnerability Rating and Scoring System" which combines "qualitative" and "quantitative" method together to calculate the vulnerability score. They uses the technique of common distribution to analyze the vulnerability rating. The medium severity vulnerabilities in VRSS should be the largest or Low ranking vulnerabilities should be smaller. VRSS is further divided into two methods i.e. qualitative rating method and Quantitative Scoring Method.

Martin et al [7] has proposed a model which provides a mechanism for scoring weakness of software. This is co-sponsored by the Software Assurance program in U.S. Department of Homeland Security (DHS). CWSS is a system which prioritizes software vulnerability in a flexible, open and consistent manner. This is also a widely accepted scoring system that addresses the needs of academia, industrialists and government. In real practice software developers face thousands of vulnerabilities and bugs that are discovered in their code.

III. CVSS (COMMON VULNERABILITY SCORING SYSTEM)

Common Vulnerability Scoring System is an open source scoring system proposed initially by National Infrastructure Assurance Council (NIAC). The first version of CVSS was released in the year 2005 (CVSS v1) while the second version with improved features was released in the year 2007 [4]. CVSS is widely adopted by security community, IT firms, and individual developers as it contains adequate information about vulnerabilities. This is also free and open to everyone for evaluating the seriousness of vulnerabilities [5,8]. CVSS gives the score to vulnerabilities on the basis of seriousness of threats and after considering several metrics. Scores are given "between" [0 to 10], 0 is given to least severe and 10 is for most severe vulnerability. A further analysis and discussion on CVSS is presented below.

CVSS has three metric groups as shown in figure 1. These groups are Base, Temporal and Environmental. Base group is used to find the score of vulnerability while the other two groups are optional for consideration.

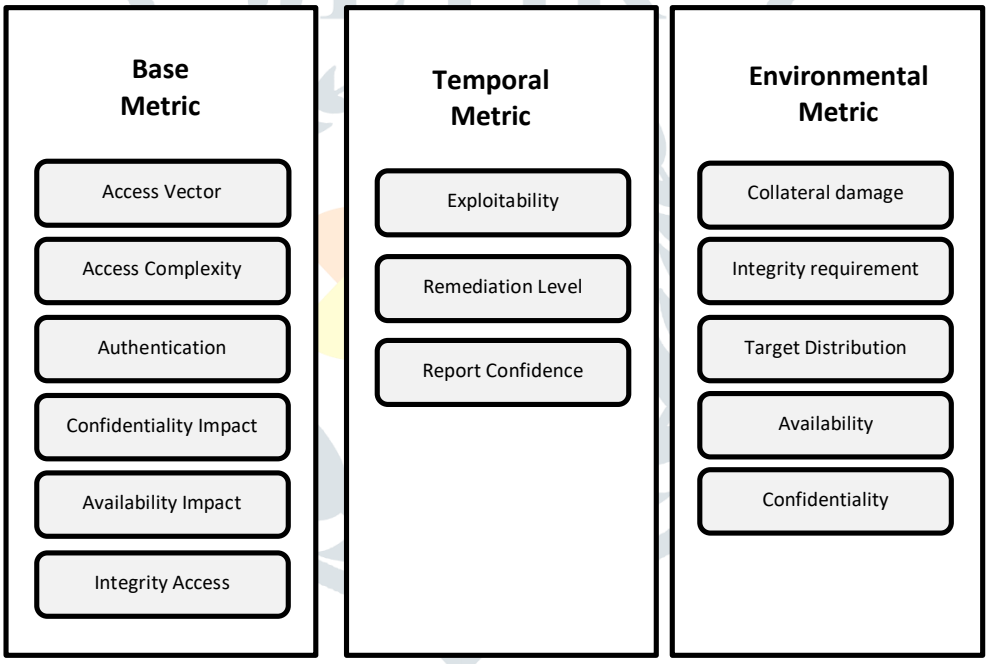

Figure 1: Metric Groups of CVSS V2

Base group further contains six metrics as given below.

I. Access Vector
II. Access Complexity
III. Authentication
IV. Confidentiality Impact
V. Integrity Impact
VI. Availability Impact.

Description of these six metrics is as follows:

Base Group Metrics:

- ***Access Vector (AV):***

Shows how the vulnerability is exploited. The more a host can be attacked, the greater is the vulnerability score. Possible metric values of access vector are Local, Network and adjacent Network.

- ***Access Complexity (AC):***

Access complexity measures the complexity of the threat required to venture the vulnerability. Buffer overflow in an Internet is the good example of this kind of attack. Metric values of access vector are High, Medium and Low.

- ***Authentication (Au)***

Authentication in CVSS calculates the no. of times an intruder must authenticate to a target to exploit vulnerability. Metric values for this metric are Multiple, Single and None.

- ***Confidentiality Impact (CI)***

Confidentiality Impact used to calculate the impact on confidentiality of an exploited flaw. Confidentiality also means preventing access by, limiting information sources and disclosure to only verified users. Metric values of CI are none, partial and complete.

- ***Integrity Impact (IC)***

Integrity Impact calculates the impact to integrity on an exploited flaw. It also refers to the guaranteed veracity and trustworthiness of information. Metric values of IC are None, Partial and Complete.

- ***Availability Impact (AI)***

Availability Impact calculates the Impact to availability of an exploited vulnerability. Availability refers to the accessibility of resources and information. Attacks that consume processor cycles, network bandwidth or disk space impact the availability of resources. Metric values are None, Partial and Complete [5].

Equation1, equation 2, equation 3 and equation 4 gives the formulas for calculation of the base score with CVSS v2. Equation 2 computes impact, Equation 3 computes exploitability and equation 4 computes f(impact). Values obtained from equation 2, equation 3 and equation 4 is used to compute the base score using equation 1in CVSS v2.

Base Score Equations:

BS= (((0.6*Impact value) + (0.4*Exploitability value)-1.5)*f(Impact)……..…(1)

Impact=10.41*(1-(1-CI)*(I-II)*(I-AI)……..(2)

Exploitability = 20 * AV * AC * Au……..……..(3)

f(impact)=0 (Zero) if impact=0 (Zero), 1.176 otherwise……(4)

IV. VRSS (VULNERABILITY RATING AND SCORING SYSTEM)

In order to lower down the risk of intruders and losses due to vulnerabilities, VRSS provides a scoring scheme which combines "qualitative" and "quantitative" method together [11]. VRSS arranges a common scheme for describing the different types of threats. Figure 2 shows the framework of VRSS.

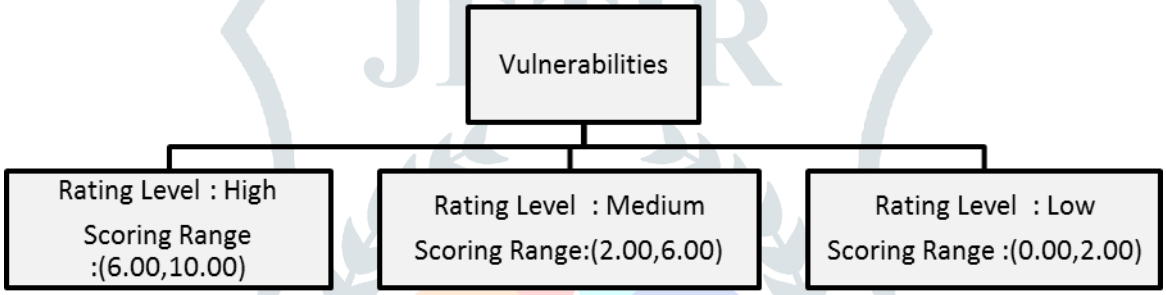

Figure 2: VRSS Framework V 1.0

VRSS uses the technique of common distribution to analyze the vulnerability rating and scoring system. The medium severity vulnerabilities in VRSS should be the largest and High or Low ranking vulnerabilities should be smaller. The main objective of system is to separate vulnerabilities from each other according to their effect. Therefore VRSS combines qualitative and quantitative methods together [6].

VRSS is further divided into two methods i.e. rating method and scoring method.

(i) Vulnerability Qualitative Rating Method

VRSS uses integrity, availability and confidentiality to access the risk level of vulnerabilities in software's. Integrity indicates the guaranteed and trustworthiness veracity of information. Availability indicates the accessibility of information and other resources. Confidentiality refers to securing information from unauthorized users and limiting information access.

Values for each of these security properties are:

- Complete
- Partial
- None

All these three metrics has three different values, so accordingly, there are 27 kinds of cases through combination of these three metrics. While rating vulnerability, direct impact is to be considered on the target host only. In case of cross site vulnerability, each of integrity, availability and confidentiality should be evaluated with a partial loss. VRSS sets the end-users as the direct target of cross site scripting vulnerability.

(ii) **Vulnerability Quantitative Scoring Method**

On the basis of Impact score calculated in vulnerability qualitative rating method, VRSS uses exploitability metrics to calculate quantitative score. Exploitability metrics is used to find out how the vulnerability is accessed. Exploitability also captures whether or not extra conditions are required to exploit the vulnerability. Equation 5 and equation 6 shows the quantitative score and exploitability score respectively.

Quantitative Score (QS)= Impact score(IS) + Exploitability score(ES)……(5)

Exploitability Score (ES)=2*Access Vector*Access Complexity*Authentication……(6)

In order to illustrate the performance VRSS analyzes all the vulnerabilities that are published between 1999 and 2008. They categorized the distribution of number of vulnerabilities in three levels i.e. high, medium and low and found the medium ranking vulnerabilities number to be the largest.

VRSS provides the different view of scoring vulnerability than CVSS as it sets end-users as the direct target of cross-site scripting vulnerabilities. In CVSS, cross site scripting is scored with partial impact to integrity and no impact to availability or confidentiality.

V. CWSS (COMMON WEAKNESS SCORING SYSTEM)

CWSS is a part of the Common Weakness Enumeration (CWE) project which provides a mechanism for scoring weakness of software. This is co-sponsored by the Software Assurance program in U.S. Department of Homeland Security (DHS) under the office of Cybersecurity and Communications. CWSS is a mechanism which prioritizes software vulnerability in a flexible, open and consistent manner. This is also a widely accepted scoring system that addresses the needs of academia, industrialists and government. In real practice software developers face thousands of vulnerabilities and bugs that are discovered in their code. Due to large number of weaknesses reported software developers are often forced to prioritize and resolve such issues [7].

Common weakness scoring system is categorized into three metric groups. These groups are as follows.

1. ***Base Finding***
2. ***Attack Surface***
3. ***Environmental***

Each group further contain factors – that are used to calculate score of weakness in software

- ***Base finding metric group:***

Base finding metric group of CWSS covers the inherent risk of the weakness, strength of control. The factors in the base finding metric group of CWSS is shown in Fig. 3.

- ***Attack Surface metric group***

Attack Surface metric of CWSS group the barriers that an intruder must overcome in order to damage the weakness. Factors in the Attack Surface metric group of CWSS are shown in Fig. 4.

- ***Environmental Metric group***

Environmental Metric group characterize the weakness that belongs to some particular environment. The factors in the Environmental Metric group of CWSS is shown in Fig. 5.

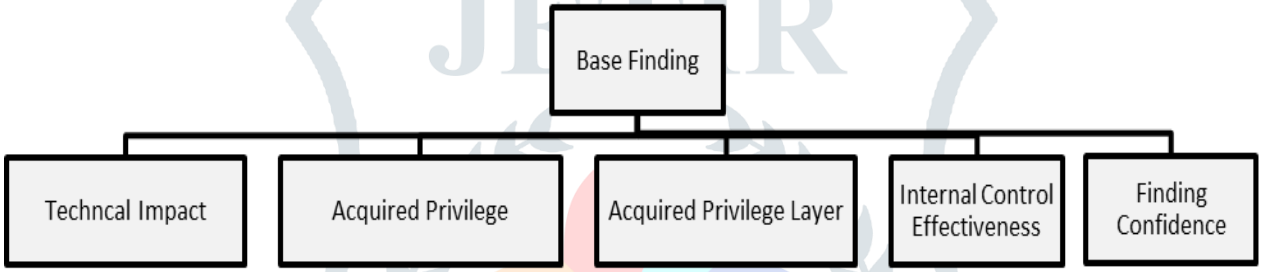

Figure 3: CWSS Base Findings Factor

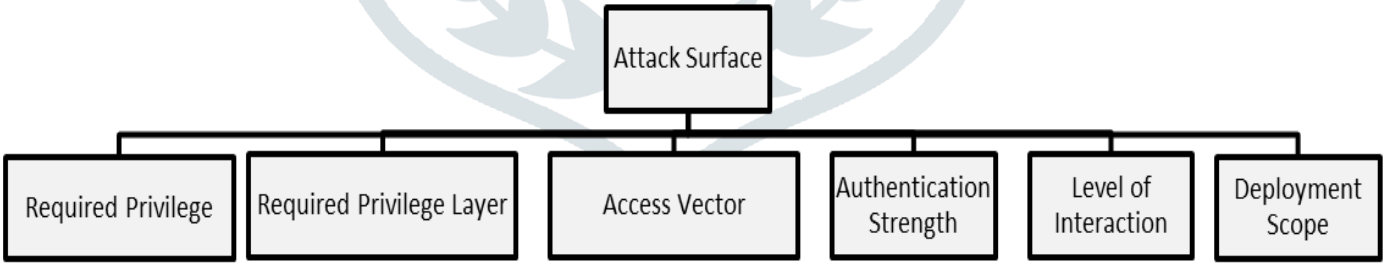

Figure 4: CWSS Attack Surface Factors

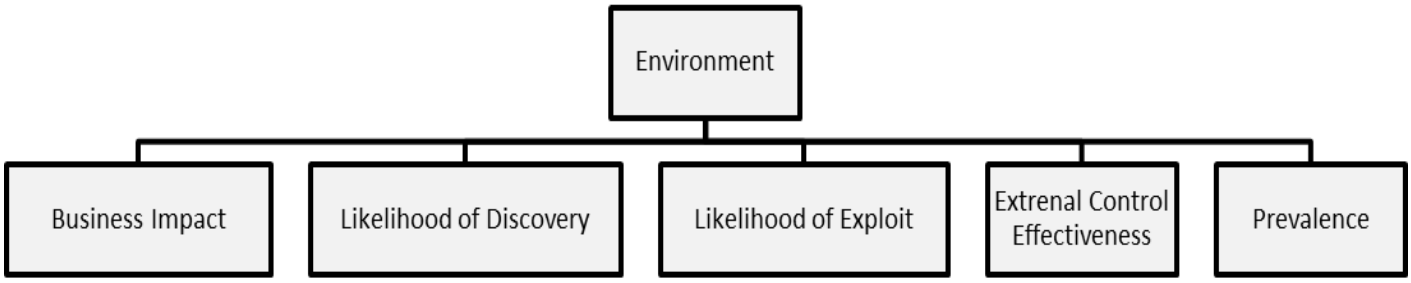

Figure 5: CWSS Environment Factors

Working of CWSS

As CWSS uses three metric groups, each factor in the groups is assigned values with associated weights and a base finding sub-score is calculated. The range of base finding sub-score is "between" (0 to 100). Their sub-score can range "between" (0 to 1). After calculation these three sub-scores are multiplied together which produces CWSS score "between" (0 to 100).

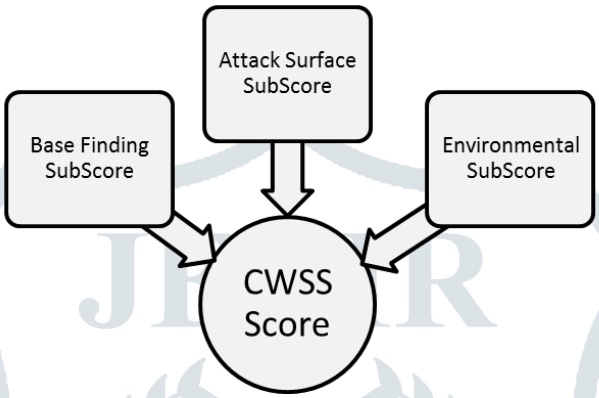

Figure 4: CWSS Score Formulation

Focus of Common Weakness Scoring System (CWSS) IS on targeted scoring and can be further extended for scoring weaknesses in a general fashion. CVSS 1.0 scoring approach could account for:

a) Likelihood of Discovery and Exploit
b) Technical Impact
c) Prevalence
d) Frequency: refers to how often a weakness occurs in a software package.

VI. CONCLUSION

In this paper, we presented a discussion on three vulnerability scoring systems i.e. (CVSS, VRSS and CWSS) to assist developers in choosing adequate scoring system according to their requirement. With people becoming dependent on technology there is also an increase in the number of software's being developed. Also, making the software's safe and secure to use by leaving no vulnerable spots is of utmost importance. Using vulnerability scoring systems can help to check the software's as how vulnerable they are. We believe that this paper will help to understand the scoring system so as to prioritize vulnerabilities.

VII. REFERENCES

[1] Somesh Mohanty "5 Important Software Vulnerabilities: Security Zone 2018.

[2] Ruyi Wang and Ling Gao/Qian Sun/Deheng Sun. An Improved CVSS-based vulnerability scoring mechanism", 2011

[3] Understanding Vulnerability Scoring to Help Measure Risk MATTHEW JERZEWSKI MAR 13, 2019

[4] Michael Shin “Threat Modeling for Security Failure-Tolerant Requirements” Social com/PASSAT/Big data 2013

[5] https://www.first.org/cvss/v2/guide

[6] Qixu Liu, Yuqing Zhang “VRSS: A new system for rating and scoring vulnerability”

[7] https://cwe.mitre.org/cwss/cwss_v1.0.1.html

[8] Peter Mell, Karen Scarfone, Sasha Romanosky “A Complete Guide to the Common Vulnerability Scoring System Version 2.0”.

[9] https://www.beyondsecurity.com/vulnerability_assessment_requirements_cvss_explained.html

[10] Ayodele Oluwasen Ibidap, Pavol Zavarsky, Dale Lindskog, Ron Ruhl. An Analysis of CVSS v2 Vulnerability Scoring.

[11] https://www.sciencedirect.com/science/article/pii/S014036641000174X