Web browser secureness with respect to CVSS

HyunChul Joh
Dept. of Computer Engineering, Kyungil University
joh@kiu.kr

Abstract

Analysis of characteristics in software vulnerabilities can be used to assess security risks and to determine the resources needed to develop patches quickly to handle vulnerability discovered. Being a new research area, the quantitative aspects of software vulnerabilities and risk assessments have not been fully investigated. However, further detailed studies are required related to the security risk assessment, using rigorous analysis of actual data which can assist decision makers to maximize the returns on their security related efforts. In this paper, quantitative software vulnerability analysis has been presented for major Web browsers (Internet Explorer (IE), Firefox (FX), Chrome (CR) and Safari (SF)) with respect to the Common Vulnerability Scoring System (CVSS). The results show that, almost all the time, vulnerabilities are compromised from remote networks with no authentication required systems, and exploitation aftermath is getting worse.

1. Introduction

These days, Web browsers are used for variety of purposes, such as personal entertainments, eLearning, online banking, or even highly confidential governmental tasks. Consequently, new vulnerabilities discovered in a Web browser put millions of the Web users at risk, requiring urgent attention from developers to address there vulnerabilities. Naturally, there has been considerable discussion of Web browser security in recent years. However, in many cases, those studies are focused on detection and prevention of individual vulnerabilities. Although quantitative data is sometimes cited, often there is no significant critical investigation.

Software vulnerabilities can be defined as software defects or weaknesses in the security system which might be exploited by malicious users causing loss or harm [1]. It is next to impossible to eliminate them completely. Those vulnerabilities are great concern since they provide attackers the ability to gain full control of the system or leakage of highly sensitive information. In this paper, we examine the secureness of the four Web browsers with respect to the CVSS. Hence, to understand this paper, background of CVSS is required, and the next section will explain it briefly.

2. CVSS Metrics

Common Vulnerability Scoring System (CVSS) is designed for providing an open and standardized way to rate mainly software related vulnerabilities [2]. This score system provides vendor independent framework for communicating the characteristics and impacts of the known vulnerabilities, and are readily available on the majority of public vulnerability databases on the Web. CVSS is composed of three metric groups: Base, Temporal and environmental. The base metric group, ranges of [0.0, 10.0], represents the intrinsic and fundamental characteristics of a vulnerability, so the score is not changed over time. The others are used to augment the base metrics and depend on the target system and changing circumstances.

The base metric has two sub-scores of exploitability and impact sub-scores. The exploitability sub-score captures how a vulnerability is accessed and whether or not extra conditions are required to exploit it while the impact sub-score measures how a vulnerability will directly affect an IT asset as the degree of losses in confidentiality, integrity, and availability.

Table I. Four Web browsers

	IE	FX	CR	SF
#of vul. †	1044	1096	979	517
Released	Aug. 1995	Nov. 2004	Sep. 2008	Jan. 2003
M.Share ‡	57.32%	18.09%	16.21%	5.67%

†number of vulnerabilities http://nvd.nist.org/ (Sep. 27th 2014)
‡market share http://www.netmarketshare.com/ (Sep. 27th 2014)

The exploitability sub-score is composed by three elements of access vector (*AV*), access complexity (*AC*), and authentication (*Au*). The access vector reflects how the vulnerability is exploited in terms of local (*L*), adjacent network (*A*), or network (*N*). The access complexity measures the complexity of the attack required to exploit the vulnerability once an attacker has gained access to the target system in terms of High (*H*), Medium (*M*), or Low (*L*). The authentication counts the number of times an attacker must authenticate to a target in order to exploit a vulnerability in terms of Multiple (*M*), Single (*S*), or None (*N*).

On the other hand, the impact sub-score is composed by the three key aspects in information security components: confidentiality, integrity and availability. The impact attributes are all assessed in terms of None (*N*), Partial (*P*), or Complete (*C*).

3. Analysis of Exploitability and Impact Sub-Scores

Table I shows the number of vulnerabilities for each Web browsers with initial release dates and market share information. Higher market share means more efficient for both black and white hats because malicious users would find it more profitable and satisfying to devote their time on software with higher market share. As a result, a smaller number of known vulnerabilities does not necessarily means a more secure software system.

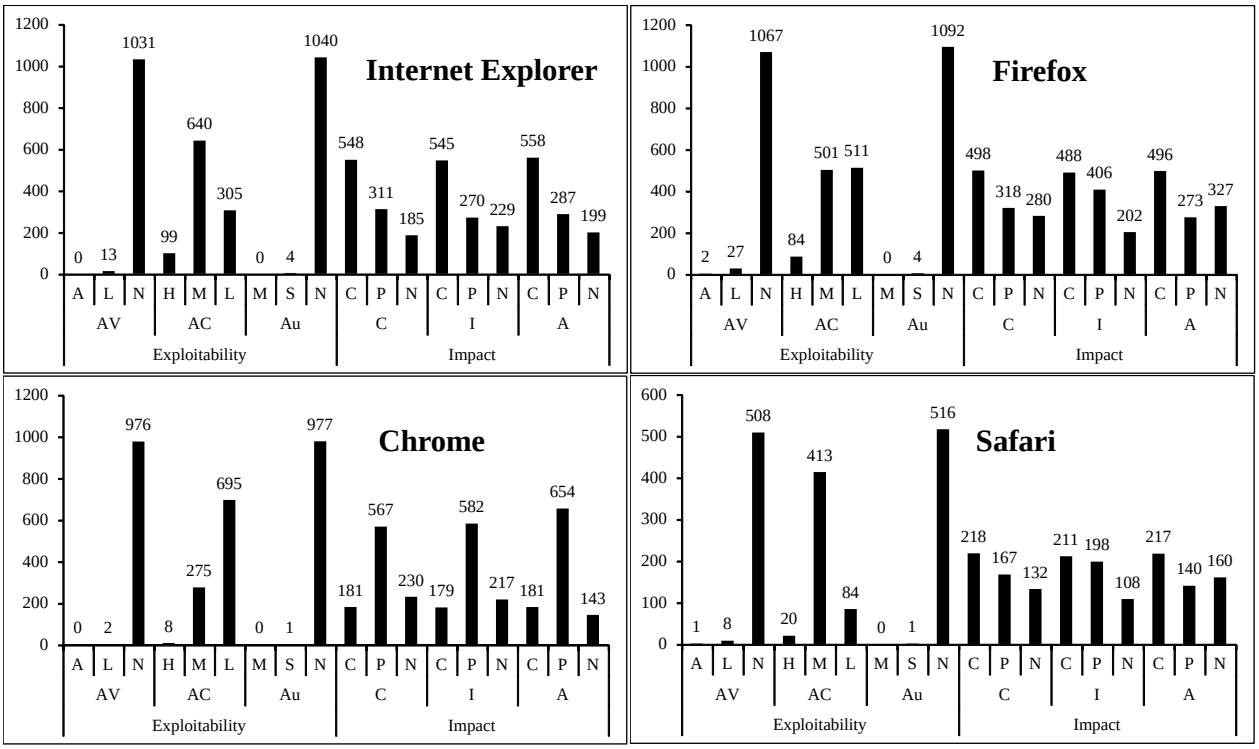

Figure 1. Number of each value in exploitability and impact sub-score groups; datasets from http://nvd.nist.org/

Table II. Top three high frequency combinations

	Freq./%†	Exploitability			Impact		
		AV	AC	Au	C	I	A
IE	498/47	N	M	N	C	C	C
	120/11	N	L	N	P	P	P
	72/6	N	L	N	N	N	P
FX	260/23	N	L	N	C	C	C
	201/18	N	M	N	C	C	C
	109/9	N	M	N	N	P	N
CR	395/40	N	L	N	P	P	P
	122/12	N	L	N	N	N	P
	102/10	N	L	N	C	C	C
SF	193/37	N	M	N	C	C	C
	82/15	N	M	N	P	P	P
	67/12	N	M	N	N	P	N

†Frequency / percentage (%)

Figure 1 shows the number of each value from CVSS in the four Web browsers. It is clearly observed that some values have more incidences than others. While Impact sub-scores relatively tend to be spread out, only some of the exploitability sub-scores have majority of the incidents.

For *AV* and *Au*, in all the Web browsers, *N*(Network) and *N*(None) have the most of numbers. This indicates that exploitations are from remote networks and if we have at least one authentication process, it is a lot safer than zero authentications. Also, in the *AC* category, the majority of them are *M* and *L*. There are very small incidents of *H*s. For the Impact sub-score, *C* (complete) takes place the highest numbers for the three categories (*C*, *I*, *A*) in IE, FX and SF whereas, interestingly, *P* (partial) is the most shown letter in CR.

Table II shows the top three individual combinations having the biggest number of vulnerabilities. For the Exploitability sub-score, vulnerabilities are occurred most of the time at <AV:N, AC:M/L, Au:N>, which means majority of systems are compromised <remotely with middle/low complexity and no authentication>. For the Impact sub-score, frequently, a compromised vulnerability let attackers completely gain IT asset in terms of CIA: the triple *C*s in the table.

4. Conclusion

This short paper analyzed the base scores from CVSS for the four most popular Web browsers quantitatively. The results show that, almost all the time, vulnerabilities are compromised from remote area at no authentication required systems. This suggests for organizations to enhance their network security-related facilities, and also to add authentication process in their systems. The result also reveals that exploitation aftermath is getting worse. An analogous study had been conducted by Scarfone and Mell [3] in 2009 based on 11,012 CVEs. They examined CVSS version 2 scoring system in depth without software categorizations.

The methods in this study do not make use of detailed information on software evolutions that may be available. Therefore, further research is needed to evaluate the impact of evolution of software products that go through many versions by explicitly considering the shared code, vulnerabilities inserted and removed in the process and the impact on resource allocation for testing and patch development.

References

[1] C. P. Pfleeger and S. L. Pfleeger, *Security in Computing*, 3rd ed. Prentice Hall PTR, 2003

[2] P. Mell, K. Scarfone and S. Romanosky, CVSS: A complete Guide to the Common Vulnerability Scoring System Version 2.0, Forum of Incident Response and Security Teams (FIRST), 2007

[3] K. Scarfone and P. Mell, An analysis of CVSS version 2 vulnerability scoring, Empirical Software Engineering and Measurement, 2009. ESEM 2009. 3rd International Symposium on , pp.516-525, 15-16 Oct. 2009