Vulnerability Trends in Web Servers and Browsers

M S Raunak
raunak@loyola.edu
Loyola University Maryland
Baltimore, MD

Richard Kuhn
kuhn@nist.gov
National Institute of Standards and Technology
Gaithersburg, MD

Richard Kogut
rwkogut@loyola.edu
Loyola University Maryland
Baltimore, MD

Raghu Kacker
National Institute of Standards and Technology
Gaithersburg, MD
raghu.kacker@nist.gov

CCS CONCEPTS

• **Software and its engineering → Software defect analysis**.

KEYWORDS

datasets, neural networks, gaze detection, text tagging

ACM Reference Format:
M S Raunak, Richard Kuhn, Richard Kogut, and Raghu Kacker. 2018. Vulnerability Trends in Web Servers and Browsers. In *HoTSoS '20: Hot Topics in the Science of Security|, Lawrence, KS.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

In previous work we have looked at trends in vulnerabilities due to ordinary programming errors. This analysis focuses on two of the most widely used types of software in today's internet, web browsers and web servers. In addition to reports of vulnerabilities, we were able to consider market share and approximate numbers of each server or browser in use, and thus able to infer some information about the impact of vulnerabilities. The key questions we sought to address are:

(1) What is the trend in vulnerabilities for these components, and the magnitude of their impact on users?
(2) Are web browsers and servers becoming more secure over time as vulnerabilities are discovered and programmers become more experienced?
(3) How do trends vary by vulnerability type?

For this analysis, we have used 2008 - 2019 data from the US National Vulnerability Database (NVD) [1]. NVD is the US government's repository of information system security vulnerabilities, which compiles nearly all publicly reported vulnerabilities using the Common Vulnerabilities and Exposures (CVE) dictionary [2]. Each reported CVE is assigned to one or more categories called the Common Weakness Enumeration (CWE) [3], which specifies categories that may include a number of subsidiary weaknesses. For example, CWE-119, Buffer errors, includes 14 subsidiary CWEs, such as out of bounds read (CWE-125), and untrusted pointer dereference (CWE-822).

By examining vulnerabilities in servers and browsers, we can gauge where the threats lie and where efforts should be addressed to prevent vulnerabilities in the future. Focusing on major internet/web applications, we considered seven web browsers and three web servers along with combinations of smaller ones. We collected the number of reported vulnerability counts in these software from year to year. Additionally, we developed a metric to gauge the relative impact of the vulnerabilities reported for the browsers and servers. To avoid any implied endorsement for a product, we have presented our findings without naming the particular software.

2 ANALYSIS AND RESULTS

The new metrics we developed include (1) raw vulnerability count and (2) market share, to compute a figure for impact score. For web browsers, the impact score trended down or began to plateau over the period studied, for most browsers with the exception of Browser-2. Raw vulnerability counts for browsers, however, varied greatly and did not show a consistent trend for all. The market share component for the metric was affected by an overall downward trend for Browser-1 and corresponding uptrend for Browser-2, whose impact can also be seen in the normalized vulnerability count graph. For other browsers, share did not vary greatly. The vulnerability count component was affected significantly by a large increase reported for buffer errors . We also looked at the proportional trend of vulnerability counts of different vulnerability types. Nearly all web browsers showed a decline in the proportion of high severity vulnerabilities with the exception of Browser-6, which is one of the newest browsers in the market. This seems to suggest that newer software may tend to be more vulnerable for attacks, but with patches and newer versions, more mature software tends to present fewer high-severity level vulnerabilities.

A similar trend analysis of web server vulnerability count and their impact shows overall gradual downward trend of their vulnerability impact with the exception of Server-2. Five categories of vulnerabilities were analyzed for the web servers, showing a gradual downward trend for most, but great variability from year to year. Vulnerability categorized as information leak seem to be showing a slight upward trend over the last couple of years. For web browsers, there persists the disappointing trend of a clearly rising buffer error type vulnerabilities over the last ten years. Overall, mature software

HoTSoS '20, April 07–08, 2020, Lawrence, KS

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/1122445.1122456

seems to withstand vulnerabilities better. These results suggest that it would be useful to place a priority on preventing buffer error type vulnerabilities in software.

ACKNOWLEDGMENTS

This work was partially supported by NIST Grant 70NANB18H278. The authors would like to thank everyone involved in maintaining the National Vulnerability Database (NVD).

REFERENCES

[1] Harold Booth, Doug Rike, and Gregory Witte. 2013. *The national vulnerability database (nvd): Overview.* Technical Report. National Institute of Standards and Technology.

[2] Peter Mell and Tim Grance. 2002. *Use of the common vulnerabilities and exposures (cve) vulnerability naming scheme.* Technical Report. NATIONAL INST OF STANDARDS AND TECHNOLOGY GAITHERSBURG MD COMPUTER SECURITY DIV.