Browser Certificate Verification and HTTPS Exception Handling

*Note: Sub-titles are not captured in Xplore and should not be used

1st Given Name Surname
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address

2nd Given Name Surname
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address

3rd Given Name Surname
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address

4th Given Name Surname
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address

5th Given Name Surname
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address

6th Given Name Surname
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address

***Abstract*—HTTPS has played a significant role in communication security by providing encryption data integrity and entity authentication. It is used hundreds of million of times every day in web browsers. Browsers report HTTPS security warnings if the certificate chain fails to validate. However, There is no clear industry consensus for browsers security strategies, and four major browsers exhibiting different warning design. To get a good understanding of the browser warnings, we design a wide variety of HTTPS certificate errors and investigate the browser warning behaviors in the field. We empirically assess whether browser security warnings are as effective as suggested by popular opinion. Our results suggest that browsers may treat the same certificate error differently. Based on our findings, we make recommendations for warning designers and researches to ensure the security of HTTPS certificate ecosystem.**

***Index Terms*—HTTPS, browser warning**

I. INTRODUCTION

HTTPS is becoming a de facto protocol used by Web to provide security features including confidentiality, data integrity, and authentication for data transmission between browsers and web servers. It is used hundreds of million of times every day in web browsers [1].

The main process of web browser certificate validation is as follows: The web server sends an endpoint certificate referencing its domain name, as well as one or more intermediate certificates to a browser. The browser must construct a trust chain from its trusted root certificates to the endpoint. Certificate validation involves checking the whole chain for that the visited domain name matches the subject in the certificate presented by the web server, that the digital signature value is valid, that all the certificates of the chain are within their validity period, that the certificates havent been revoked, that various extensions are meet requirements, and many other checks. If that validation fails, browsers display HTTPS security warnings to warn users of potential network attacks. However, if warnings cannot communicate risks correctly, uses may make wrong decisions.

HTTPS is a highly complex protocol, and different web browsers include their own, proprietary implementations.

Our work focuses on the X.509 certificate validation rather than the whole process of HTTPS implements, but we hope our work can be useful for deploying HTTPS faster and better.

We make the following contributions:

- We make various HTTPS errors including error about HSTS and HPKP policy, and record the corresponding warning behaviors of major browsers.
- We find that some browsers still trust certificates with well-known security risks, for example, IE doesnt provide any warning for MD5 certificates, which are known to be vulnerable to prefix-collision attacks [2].
- We compare the major browsers' different warning behaviors for the same HTTPS error. Additionally, we accordingly make recommendations for browser warning to convey an appropriate level of risk.

II. RELATED WORK

We are not aware of any prior work on discovery of risk level vulnerabilities in the browser HTTPS security warnings. However, previous studies have shown the need for that research to improve HTTPS implementations and browser warnings.

Researchers take large scale scans of HTTPS certificate ecosystem including IPv4 space, certificate transparency logs, and Alexa Top 1 Million websites. As a result, they find a large proportion of invalid certificate in the wild [3]. More

Identify applicable funding agency here. If none, delete this.

than that, there are even many invalid certificates signed by browser-trusted CAs [4].

III. HTTPS ERRORS IN BROWSERS

A. Browser Security Warnings

Browser uses security warnings to protect users by alerting them to network attacks. According to different levels of risk, Browser presents different connection security states in order to maximize security and minimize side effects of user experience. For low-risk errors, it will show a passive security indicator without interrupting the users browsing. For medium-risk errors, it will show a bypassable warning that discourages the user from continuing. For high-risk errors, it will show a security warning that the user cannot bypass. We describe security warnings of major browsers.

1) Risk Level A: Low-risk warnings: In this scenario, a passive security indicator indicates minor HTTPS errors by removing lock icon, changing lock icons color, providing textual information, or by other means without interrupting the users browsing. this is probably because webmaster has

Fig. 1. low-risk warning in chrome.

configured his website correctly, including purchasing a valid certificate, but the website still contains images or scripts over plain HTTP. Although the browser was able to establish a valid HTTPS connection, there are still minor problems.

2) Risk Level B: Medium-risk warnings that can be bypassed: Bypassable HTTPS security warnings are the most common type of browser warning. Browsers use bypassable HTTPS security warning to treat the vast majority of HTTPS errors. In this scenario, if there is an HTTPS error, the

Fig. 2. medium-risk warning in chrome.

browser will stop the page load and display an HTTPS security warning. Typically, users are able to click through the warning by clicking on a button, but this may be disabled if the website servers the HTTP Strict Transport Security (HSTS) or HTTP Public Key Pinning (HPKP) header. Although the HTTP(s) traffic we captured through Fiddler[https://www.telerik.com/fiddler] shows the web page is not transmitted in plain text, clicking through the warning may allow an actual man-in-the-middle attack to proceed.

3) Risk Level C: High-risk warnings that cannot be bypassed: As a worst case scenario, the browser will prevent uses from continuing to browse the website by showing a security warning that the users cannot bypass. this is probably

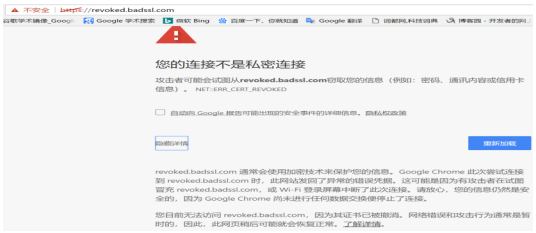

Fig. 3. high-risk warning in chrome.

because the website is supposed to establish a valid HTTPS connection, but the certificate chain fails to validate (e.g., a HTTPS certificate error occurs in a website that has deployed HSTS policy).

4) Cannot establish HTTPS connection: The browser may fail to establish a HTTPS connection to the server for unsupported TLS version, cipher mismatch, or other reasons. In this scenario, the browser cannot obtain certificates from the web server, let alone certificate validation, so the TLS handshake error is not discussed in this paper.

Fig. 4. Cannot establish HTTPS connection in chrome.

REFERENCES

[1] A. P. Felt, R. Barnes, A. King, C. Palmer, C. Bentzel, and P. Tabriz, "Measuring https adoption on the web," in *26th USENIX Security Symposium*, 2017, pp. 1323–1338.

[2] M. Stevens, A. Sotirov, J. Appelbaum, A. Lenstra, D. Molnar, D. A. Osvik, and B. De Weger, "Short chosen-prefix collisions for md5 and the creation of a rogue ca certificate," in *Advances in Cryptology-CRYPTO 2009*. Springer, 2009, pp. 55–69.

[3] R. Holz, L. Braun, N. Kammenhuber, and G. Carle, "The ssl landscape: a thorough analysis of the x. 509 pki using active and passive measurements," in *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*. ACM, 2011, pp. 427–444.

[4] Z. Durumeric, J. Kasten, M. Bailey, and J. A. Halderman, "Analysis of the https certificate ecosystem," in *Proceedings of the 2013 conference on Internet measurement conference*. ACM, 2013, pp. 291–304.