Usability of TLS/PKI

Mainack Mondal

CS 60081
Autumn 2021
Roadmap

- Trust on the web
  - SSL notifications
SSL/ TLS

- Secure Sockets Layer (SSL) and its successor, Transport Layer Security (TLS)
  - Enable secure communication

- Frequently encountered with web browsing (HTTPS) and more behind the scenes protocols
  - VOIP, etc
Attacker Model

- Passive: People eavesdropping
  - Goal: content of what we are sending
  - Goal: getting session cookies

- Active: Man in the middle attack
  - Goal: intercept all communication
  - Solution: authenticate that we are talking to the right site, not an imposter
  - Solution: Use certificates inside a public-key infrastructure (PKI)
Certificates → Trust

- Public-Key Infrastructure (PKI)
  - Certificates are issued by certificate authorities that bind cryptographic keys to identities
  - In https:
    - Identifies == web site owners (google.com)
    - By Certificate Authority (CA) == GlobalSign
    - Using Certificates: proving authenticity of google.com which is verified by Globabsign
Certificate: point of view of users

Different for different browsers
Usability: The look of SSL is non-uniform across browsers

<table>
<thead>
<tr>
<th>Browser</th>
<th>HTTPS</th>
<th>HTTPS minor error</th>
<th>HTTPS major error</th>
<th>HTTP</th>
<th>EV</th>
<th>Malware</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edge 20 Win</td>
<td><a href="https://example.com">https://example.com</a></td>
<td><a href="https://mix">https://mix</a></td>
<td></td>
<td>example.com</td>
<td>Symantec Co</td>
<td>[x] Unsafe website</td>
</tr>
<tr>
<td>Opera Mini 14 And</td>
<td>[x] <a href="http://www.example">www.example</a></td>
<td>mixed.badsssl.c</td>
<td>wrong.host.ba</td>
<td>[x] <a href="http://www.example">www.example</a></td>
<td>[x] Symantec Corp</td>
<td>[x] [x] Blocked</td>
</tr>
<tr>
<td>UC Mini 10 And</td>
<td>Example Di</td>
<td>mixed.bads</td>
<td>Blocked</td>
<td>Example Di</td>
<td>[x] Endpoint, C.</td>
<td>[x] [x] Blocked</td>
</tr>
<tr>
<td>UC Browser 2 iOS</td>
<td>Example Di</td>
<td>mixed.bads</td>
<td>wrong.host</td>
<td>Example Di</td>
<td>[x] Endpoint, C.</td>
<td>[x] [x] Blocked</td>
</tr>
<tr>
<td>Safari 9 iOS</td>
<td>example.com</td>
<td>mixed.badssl.s</td>
<td>wrong.host</td>
<td>example.com</td>
<td>Symantec</td>
<td>[x] [x] Unavailable</td>
</tr>
</tbody>
</table>

Figure 2: Security indicators for major browsers on Windows (Win), Mac, Android (And), and iOS. For categories that trigger warnings (e.g., malware), we include the security indicator state during the warning.

Felt et al., SOUPS 2016, required reading
PKI: Point of view of browser software

- Couple of hundreds of trusted CAs
  - Hardcoded into browsers
  - They issue certificates ➔ Certificate authorities (CAs) sign the certificates binding identities (domain name) to keys
  - These keys are used for proving identity + secure communication
PKI: Point of view of domain owners

- Apply for a certificate to a CA
  - Validation process + payment
  - Give you a certificate file
  - You put that file in your server
Security/Usability issues

- Implementation issues
- Compromised CA
- Man-in-the-middle attacks
  - Downgrade/dumbing-down attacks
  - Addition of “rogue” certificates
- Revocation of certificates
- Timing attacks and other side channels
- Communicating to users the underlying mechanism / issues
Example of implementation issue

- Heartbleed: OpenSSL bug
  - CVE-2014-0160
  - TLS heartbeat extension misses a bounds check and thus lets an attacker “read” memory
Compromised CAs (yes it happens)

- Comodo and Diginotar suffered breaches in 2011 that let attackers issue rogue certificates which the attackers know

- Untrustworthy CAs
  - Compelled certificate creation attacks
  - “This paper introduces the compelled certificate creation attack, in which government agencies may compel a certificate authority to issue false SSL certificates that can be used by intelligence agencies to covertly intercept and hijack individuals’ secure Web-based communications”
Man-in-the-middle attacks (MITM)

- many corporations perform MITM attacks by adding certificates to users’ computers
  - Lenovo PCs ship with man-in-the-middle adware that breaks HTTPS connections: https://arstechnica.com/?post_type=post&p=615263
- A man in the middle can also tell you a site doesn’t support SSL/TLS (downgrade) or any strong ciphers (dumbing down)
Validity of a certificate

- Certificates can be revoked in case of a compromise

- Certificate Revocation Lists (CRLs) were used, but they got really large – Incremental updates were better

- Online Certificate Status Protocol (OCSP)
Self signed certificates

• You can create a certificate yourself
  • A good certificate: CA signs the certificate
  • A bad certificate: you sign it yourself OR use certificate meant for someone else

• Naturally the browser complains
  • You get a warning
  • Example: https://grey-dev.ece.cmu.edu/
This Connection Is Not Private

This website may be impersonating “grey-dev.ece.cmu.edu” to steal your personal or financial information. You should go back to the previous page.
This Connection Is Not Private

This website may be impersonating “grey-dev.ece.cmu.edu” to steal your personal or financial information. You should go back to the previous page.

Safari warns you when a website has a certificate that is not valid. This may happen if the website is misconfigured or an attacker has compromised your connection.

To learn more, you can view the certificate. If you understand the risks involved, you can visit this website.
Your connection is not private

Attackers might be trying to steal your information from grey-dev.ece.cmu.edu (for example, passwords, messages, or credit cards). Learn more

NET::ERR_CERT_COMMON_NAME_INVALID

Help improve Chrome security by sending URLs of some pages you visit, limited system information, and some page content to Google. Privacy policy

Advanced

Back to safety
Your connection is not private

Attackers might be trying to steal your information from grey-dev.ece.cmu.edu (for example, passwords, messages, or credit cards). Learn more

NET::ERR_CERT_COMMON_NAME_INVALID

☐ Help improve Chrome security by sending URLs of some pages you visit, limited system information, and some page content to Google. Privacy policy

Hide advanced  Back to safety

This server could not prove that it is grey-dev.ece.cmu.edu; its security certificate is from grey-dev.andrew.cmu.edu. This may be caused by a misconfiguration or an attacker intercepting your connection.

Proceed to grey-dev.ece.cmu.edu (unsafe)
Warning: Potential Security Risk Ahead

Firefox detected a potential security threat and did not continue to grey-dev.ece.cmu.edu. If you visit this site, attackers could try to steal information like your passwords, emails, or credit card details.

What can you do about it?

The issue is most likely with the website, and there is nothing you can do to resolve it. You can notify the website’s administrator about the problem.

Learn more...

Go Back (Recommended)    Advanced...
Firefox detected a potential security threat and did not continue to grey-dev.ece.cmu.edu. If you visit this site, attackers could try to steal information like your passwords, emails, or credit card details.

What can you do about it?
The issue is most likely with the website, and there is nothing you can do to resolve it. You can notify the website's administrator about the problem.

Learn more...

Websites prove their identity via certificates. Firefox does not trust this site because it uses a certificate that is not valid for grey-dev.ece.cmu.edu. The certificate is only valid for grey-dev.andrew.cmu.edu.

Error code: **SSL_ERROR_BAD_CERT_DOMAIN**

View Certificate

Go Back (Recommended)  Accept the Risk and Continue
Which warning prompt ensures security of the user?
Rethinking Connection Security Indicators

Adrienne Porter Felt¹, Robert W. Reeder¹, Alex Ainslie¹, Helen Harris¹, Max Walker¹, Christopher Thompson², Mustafa Emre Acer¹, Elisabeth Morant¹, Sunny Consolvo¹
Google¹, UC Berkeley²
security-enamel@chromium.org¹, cthompson@cs.berkeley.edu²

ABSTRACT
We propose a new set of browser security indicators, based on user research and an understanding of the design challenges faced by browsers. To motivate the need for new security indicators, we critique existing browser security indicators and survey 1,329 people about Google Chrome’s indicators. We then evaluate forty icons and seven complementary strings by surveying thousands of respondents about their perceptions of the candidates. Ultimately, we select and propose three indicators. Our proposed indicators have been adopted by Google Chrome, and we hope to motivate others to update their security indicators as well.

Figure 1: The green lock is a security indicator.

We then began the task of creating and testing new security indicators, working within the additional constraints posed by modern browser needs. Browsers are used by diverse audiences on diverse devices. Security indicators therefore face several design constraints:
Felt et al. (Required reading)

- survey 1,329 people about Google Chrome’s indicators using a Google chrome extension
- screenshot of Chrome’s URL bar with a red circle around the lock icon
  - What does the green symbol to the left of the URL mean to you?
- Coding (codebook development)
  - One codemaster used open coding for an initial codebook
  - Two partial coding rounds: each time giving feedback to the codemaster about shortcomings in the codebook