

CSE 484: Computer Security and Privacy

Usability Side-Channels

Spring 2023

David Kohlbrenner

dkohlbre@cs

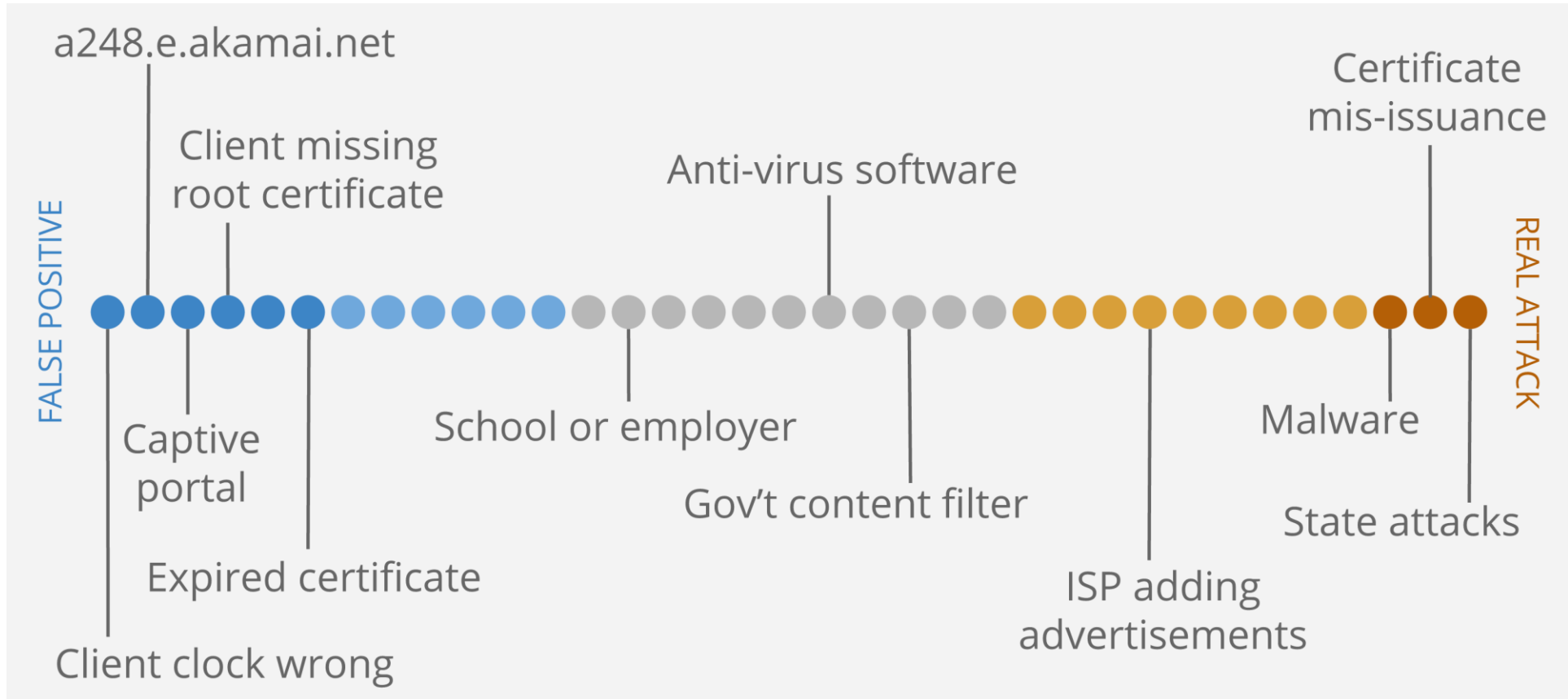
Thanks to Franzi Roesner, Dan Boneh, Dieter Gollmann, Dan Halperin, David Kohlbrenner, Yoshi Kohno, Ada Lerner, John Manferdelli, John Mitchell, Vitaly Shmatikov, Bennet Yee, and many others for sample slides and materials ...

Logistics

- Monday is a guest lecture, not recorded!
 - Come and ask questions, it'll be interesting stuff
- Next Friday is not recorded either, and is also an interesting topic
- No class Wednesday (24th)
 - I will have to cancel my office hours as well
- Lab 3 (patch for exploit1) due Monday
- FP part 1 (RCAs for 2 of exploit{2,3,4}) due in a week+

Usability and Security - Warnings

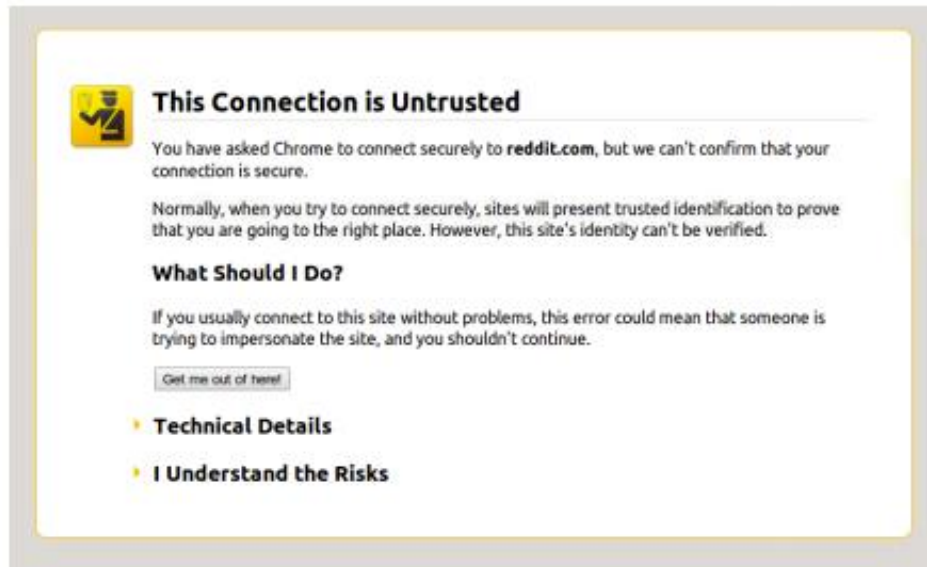
Challenge: Meaningful Warnings



See current designs for different conditions at <https://badssl.com/>.

Firefox vs. Chrome Warning

33% vs. 70% clickthrough rate



This Connection is Untrusted

You have asked Chrome to connect securely to **reddit.com**, but we can't confirm that your connection is secure.

Normally, when you try to connect securely, sites will present trusted identification to prove that you are going to the right place. However, this site's identity can't be verified.

What Should I Do?

If you usually connect to this site without problems, this error could mean that someone is trying to impersonate the site, and you shouldn't continue.

[Get me out of here!](#)

- Technical Details
- I Understand the Risks



This is probably not the site you are looking for!

You attempted to reach **reddit.com**, but instead you actually reached a server identifying itself as **a248.e.akamai.net**. This may be caused by a misconfiguration on the server or by something more serious. An attacker on your network could be trying to get you to visit a fake (and potentially harmful) version of **reddit.com**.

You should not proceed, **especially** if you have never seen this warning before for this site.

[Proceed anyway](#) [Back to safety](#)

[Help me understand](#)

Experimenting w/ Warning Design

#	Condition	CTR	N
1	Control (default Chrome warning)		
2	Chrome warning with policeman		
3	Chrome warning with criminal		
4	Chrome warning with traffic light		
5	Mock Firefox		
6	Mock Firefox, no image		
7	Mock Firefox with corporate styling		

Table 1. Click-through rates and sample size for conditions.

Experimenting w/ Warning Design

#	Condition	CTR	N
1	Control (default Chrome warning)	67.9%	17,479
2	Chrome warning with policeman		
3	Chrome warning with criminal		
4	Chrome warning with traffic light		
5	Mock Firefox		
6	Mock Firefox, no image		
7	Mock Firefox with corporate styling		

Table 1. Click-through rates and sample size for conditions.

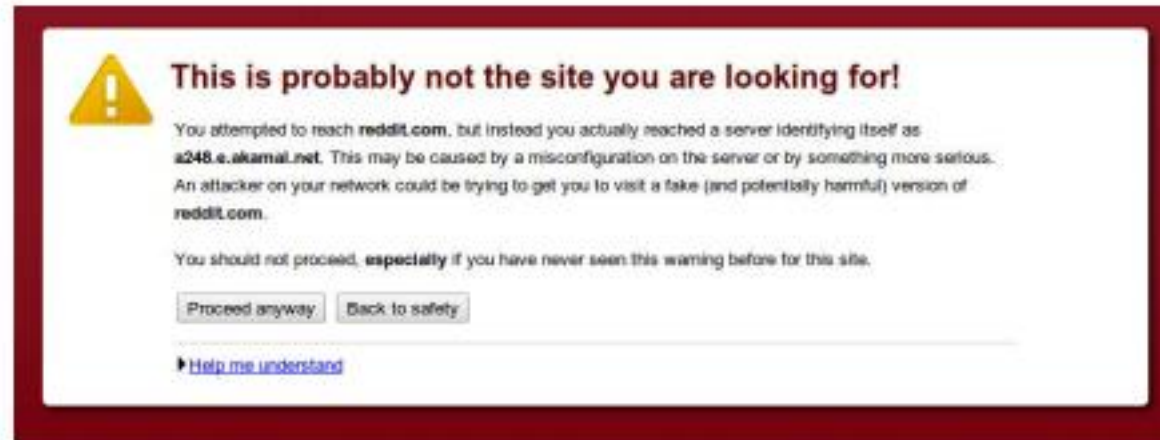


Figure 1. The default Chrome SSL warning (Condition 1).

Experimenting w/ Warning Design

#	Condition	CTR	N
1	Control (default Chrome warning)	67.9%	17,479
2	Chrome warning with policeman	68.9%	17,977
3	Chrome warning with criminal	66.5%	18,049
4	Chrome warning with traffic light	68.8%	18,084
5	Mock Firefox		
6	Mock Firefox, no image		
7	Mock Firefox with corporate styling		

Table 1. Click-through rates and sample size for conditions.



Figure 1. The default Chrome SSL warning (Condition 1).

Figure 4. The three images used in Conditions 2-4.

Experimenting w/ Warning Design

#	Condition	CTR	N
1	Control (default Chrome warning)	67.9%	17,479
2	Chrome warning with policeman	68.9%	17,977
3	Chrome warning with criminal	66.5%	18,049
4	Chrome warning with traffic light	68.8%	18,084
5	Mock Firefox	56.1%	20,023
6	Mock Firefox, no image	55.9%	19,297
7	Mock Firefox with corporate styling		

Table 1. Click-through rates and sample size for conditions.

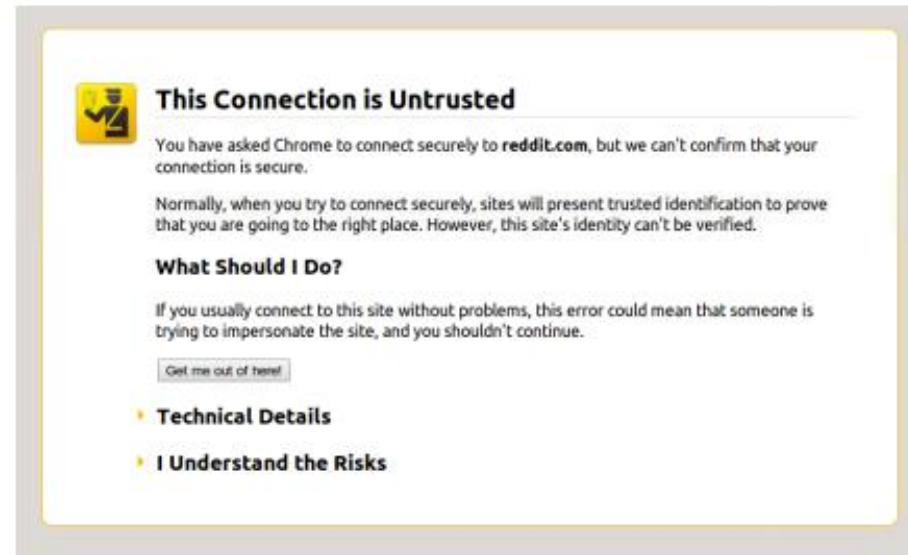


Figure 2. The mock Firefox SSL warning (Condition 5).

Experimenting w/ Warning Design

#	Condition	CTR	N
1	Control (default Chrome warning)	67.9%	17,479
2	Chrome warning with policeman	68.9%	17,977
3	Chrome warning with criminal	66.5%	18,049
4	Chrome warning with traffic light	68.8%	18,084
5	Mock Firefox	56.1%	20,023
6	Mock Firefox, no image	55.9%	19,297
7	Mock Firefox with corporate styling	55.8%	19,845

Table 1. Click-through rates and sample size for conditions.

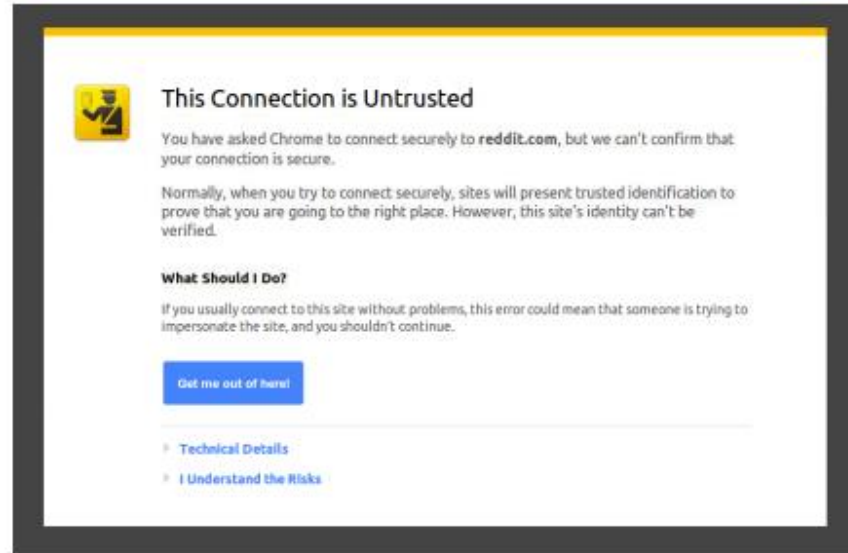
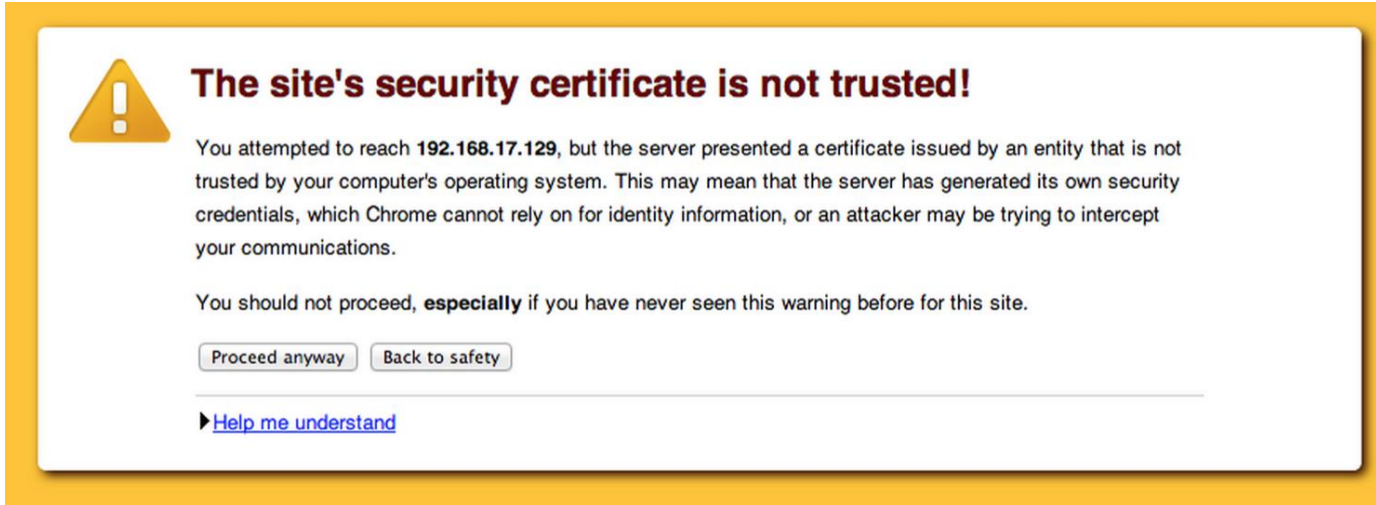



Figure 3. The Firefox SSL warning with Google styling (Condition 7).

Opinionated Design Helps!



 **The site's security certificate is not trusted!**

You attempted to reach **192.168.17.129**, but the server presented a certificate issued by an entity that is not trusted by your computer's operating system. This may mean that the server has generated its own security credentials, which Chrome cannot rely on for identity information, or an attacker may be trying to intercept your communications.

You should not proceed, **especially** if you have never seen this warning before for this site.

[▶ Help me understand](#)

Adherence	N
30.9%	4,551

Opinionated Design Helps!

The site's security certificate is not trusted!

You attempted to reach **192.168.17.129**, but the server presented a certificate not trusted by your computer's operating system. This may mean that the server's identity is not what it claims to be, or that the server's credentials, which Chrome cannot rely on for identity information, or an attacker is intercepting your communications.

You should not proceed, **especially** if you have never seen this warning before.

[Proceed anyway](#) [Back to safety](#)

[Help me understand](#)

Your connection is not private

Attackers might be trying to steal your information from **reddit.com** (for example, passwords, messages, or credit cards).

[Proceed to the site \(unsafe\)](#) [Back to safety](#)

[Advanced](#)

Your connection is not private

Attackers might be trying to steal your information from **www.example.com** (for example, passwords, messages, or credit cards).

[Advanced](#) [Back to safety](#)

Adherence	N
30.9%	4,551
32.1%	4,075
58.3%	4,644

Today's warnings (2022)

Deprecated encryption schemes



This site can't provide a secure connection

rc4.badssl.com uses an unsupported protocol.

ERR_SSL_VERSION_OR_CIPHER_MISMATCH

Details



5/19/2023

Secure Connection Failed

An error occurred during a connection to rc4.badssl.com. Cannot communicate securely with peer: no common encryption algorithm(s).

Error code: SSL_ERROR_NO_CIPHER_OVERLAP

- The page you are trying to view cannot be shown because the authenticity of the received data could not be verified.
- Please contact the website owners to inform them of this problem.

[Learn more...](#)

Try Again



CSE 484 - Spring 2023

14

Expired certificates



Your connection is not private

Attackers might be trying to steal your information from **expired.badssl.com** (for example, passwords, messages, or credit cards). [Learn more](#)

NET::ERR_CERT_DATE_INVALID

💡 To get Chrome's highest level of security, [turn on enhanced protection](#)

Advanced

Back to safety



5/19/2023



Warning: Potential Security Risk Ahead

Firefox detected an issue and did not continue to expired.badssl.com. The website is either misconfigured or your computer clock is set to the wrong time.

It's likely the website's certificate is expired, which prevents Firefox from connecting securely. 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?

Your computer clock is set to 12/7/2022. Make sure your computer is set to the correct date, time, and time zone in your system settings, and then refresh expired.badssl.com.

If your clock is already set to the right time, the website is likely misconfigured, and there is nothing you can do to resolve the issue. You can notify the website's administrator about the problem.

[Learn more...](#)

Go Back (Recommended)

Advanced...



CSE 484 - Spring 2023

15

Self-signed certificates



Your connection is not private

Attackers might be trying to steal your information from **self-signed.badssl.com** (for example, passwords, messages, or credit cards). [Learn more](#)

NET::ERR_CERT_AUTHORITY_INVALID

💡 To get Chrome's highest level of security, [turn on enhanced protection](#)

Advanced

Back to safety



5/19/2023



Warning: Potential Security Risk Ahead

Firefox detected a potential security threat and did not continue to self-signed.badssl.com. If you visit this site, attackers could try to steal information like your passwords, emails, or credit card details.

[Learn more...](#)

Go Back (Recommended)

Advanced...



CSE 484 - Spring 2023

16

Untrusted Root certificate



Your connection is not private

Attackers might be trying to steal your information from **untrusted-root.badssl.com** (for example, passwords, messages, or credit cards). [Learn more](#)

NET::ERR_CERT_AUTHORITY_INVALID

💡 To get Chrome's highest level of security, [turn on enhanced protection](#)

Advanced

Back to safety



5/19/2023



Warning: Potential Security Risk Ahead

Firefox detected a potential security threat and did not continue to untrusted-root.badssl.com. 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.

If you are on a corporate network or using anti-virus software, you can reach out to the support teams for assistance. You can also notify the website's administrator about the problem.

[Learn more...](#)

Go Back (Recommended)

Advanced...



CSE 484 - Spring 2023

17

Address Bar behaviors (2022)



⚠ Not secure | ~~https~~://self-signed.badssl.com



🔒 Not Secure https://self-signed.badssl.com

Does anything stand out?

- Canvas
- What makes warnings hard, especially over time?
- Why do Firefox and Chrome make different warning designs?



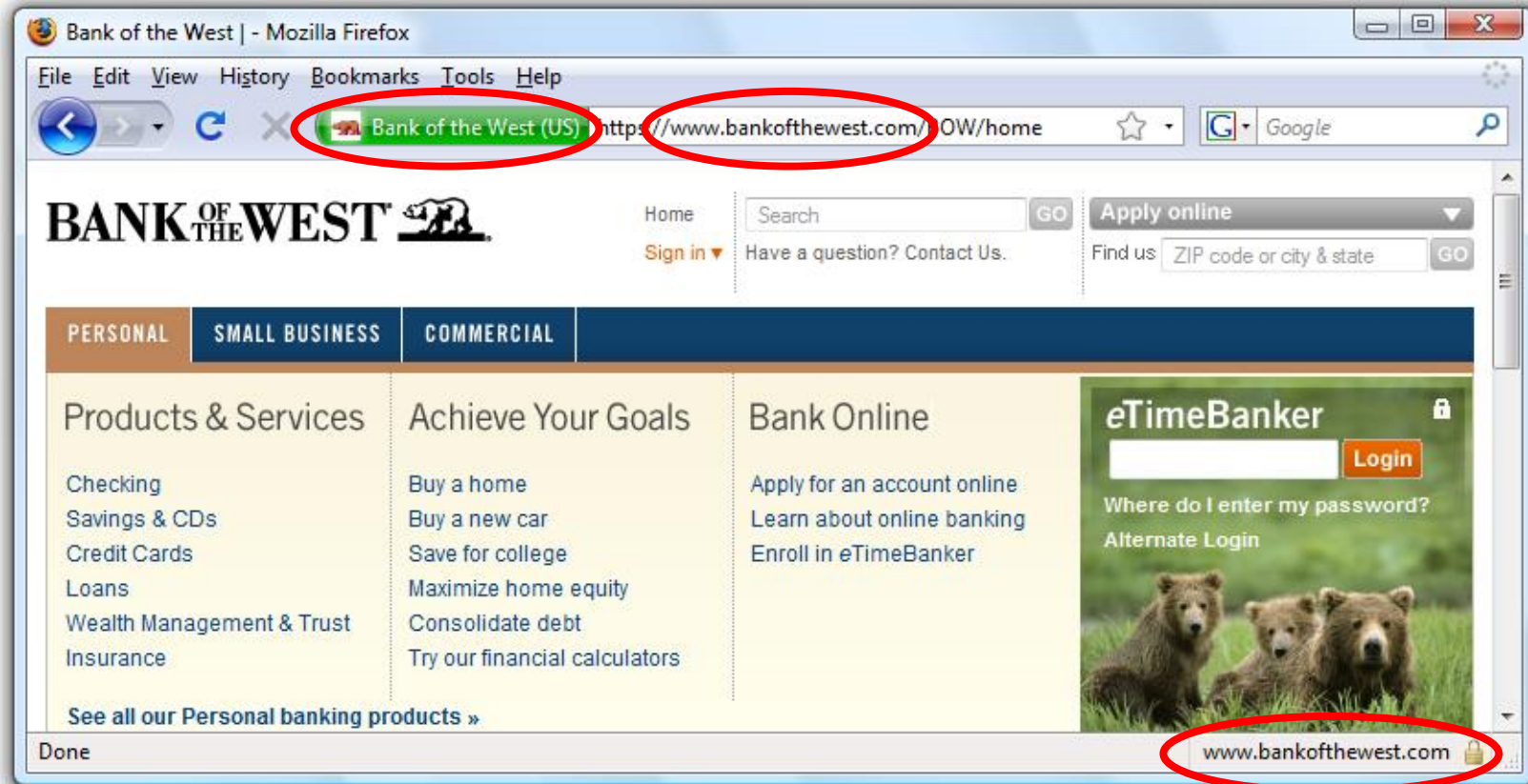
Case Study #2: Phishing

- **Design question:** How do you help users avoid falling for phishing sites?

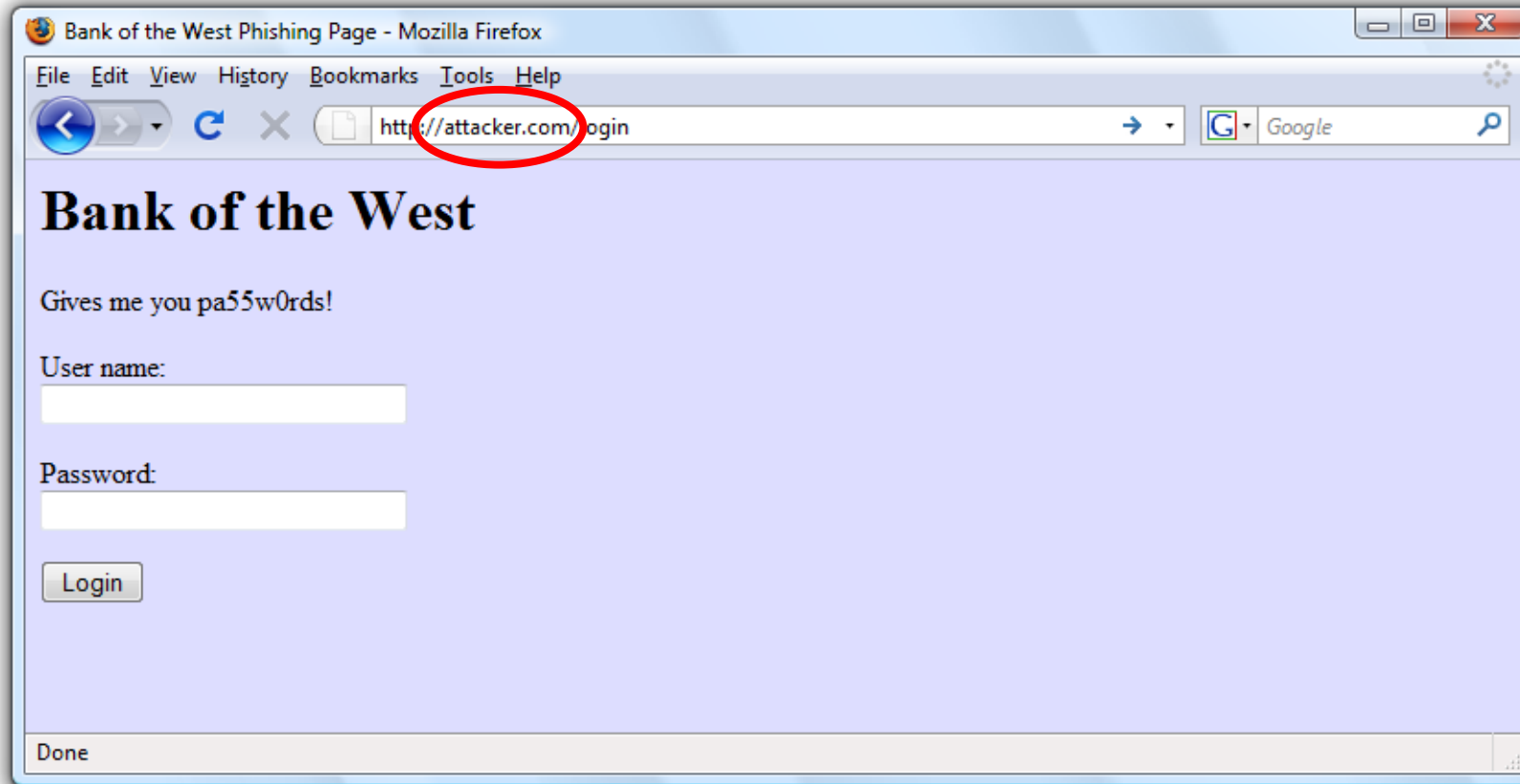
A Typical Phishing Page

The screenshot shows a browser window titled "PayPal - Welcome". The address bar contains the URL `http://www.ipaypal.szm.sk/login.html`, which is circled in red. A red box highlights this URL with the text "Weird URL http instead of https". The page features the PayPal logo, navigation tabs for "Welcome", "Send", and "Auction Tools", and a "Member Log-In" section with input fields for "Email Address" and "Password". Other elements include a "Join PayPal Today" button, a "Shop Without Sharing" banner, and various promotional tiles for "Buyers", "eBay Sellers", and "Merchants".

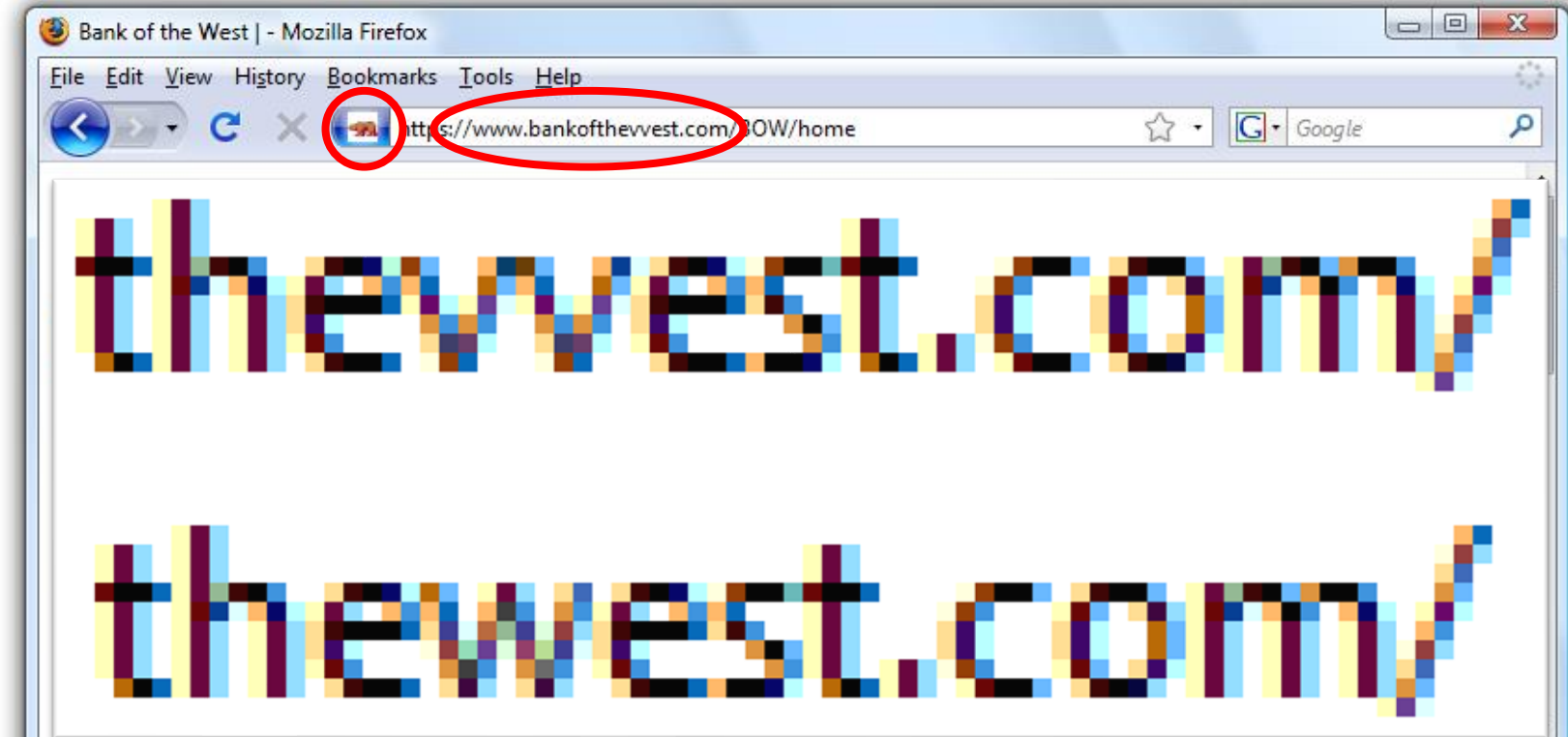
Safe to Type Your Password?



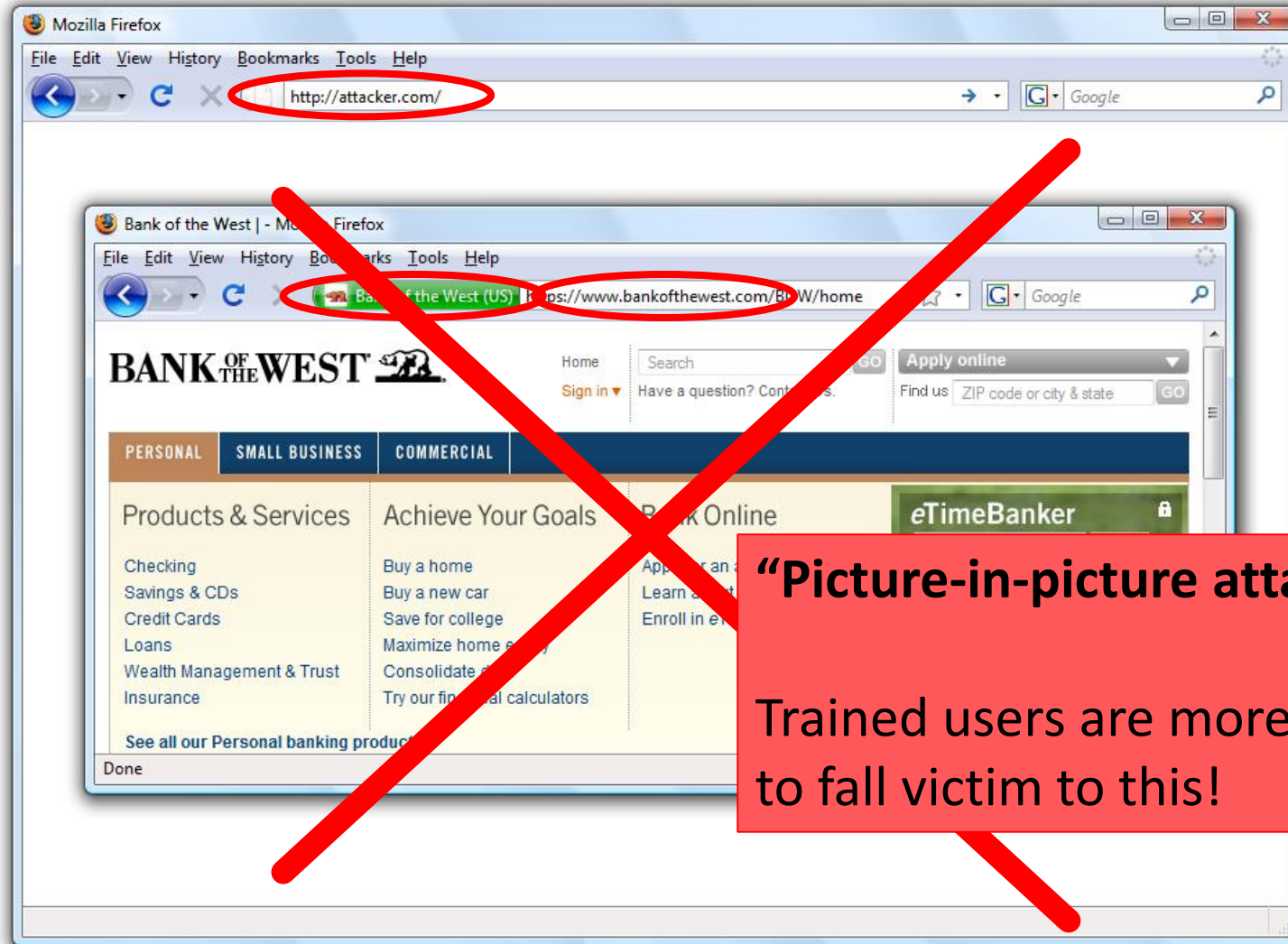
Safe to Type Your Password?



Safe to Type Your Password?

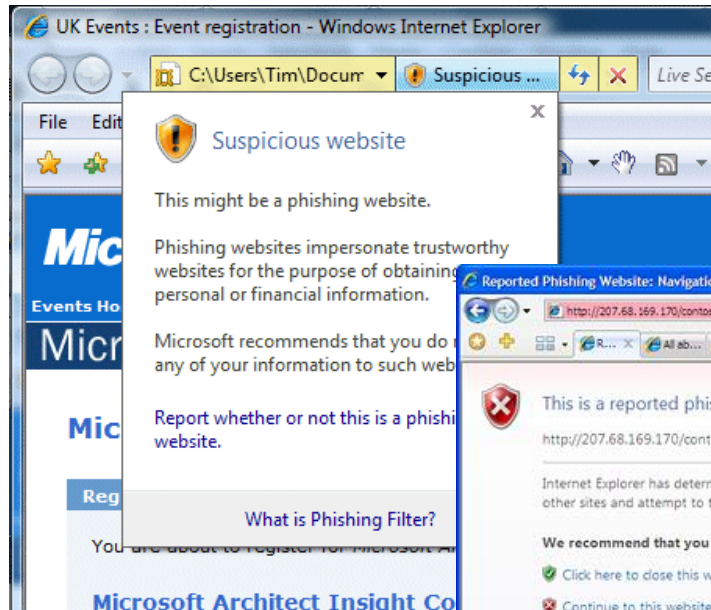


Safe to Type Your Password?

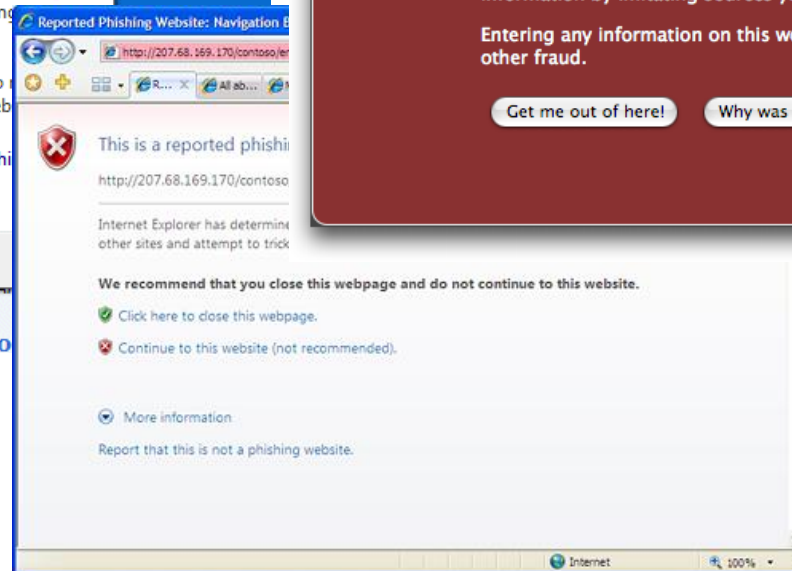


“Picture-in-picture attacks”
Trained users are more likely to fall victim to this!

Phishing Warnings (2008)



Passive (IE)



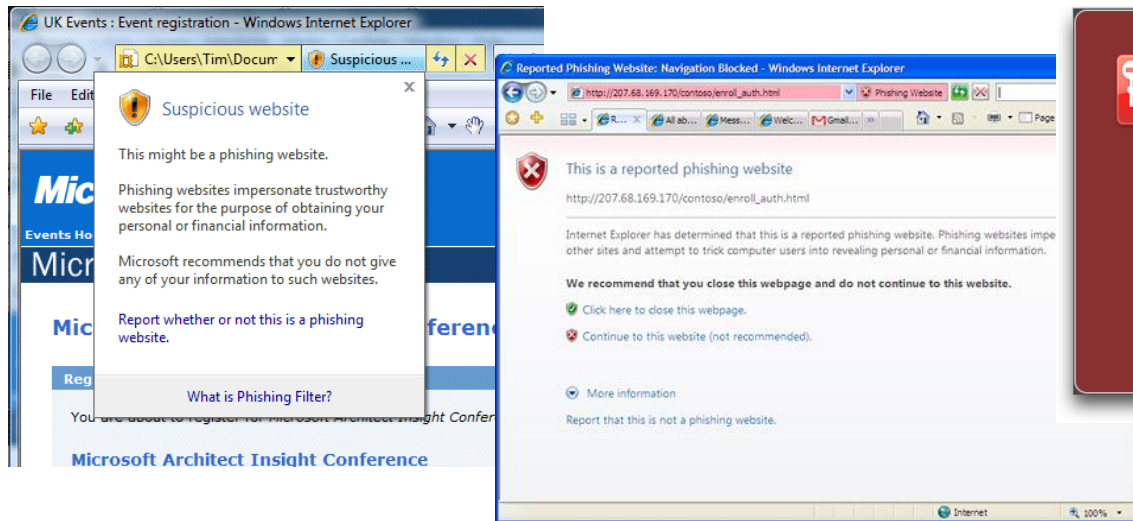
Active (IE)



Active (Firefox)

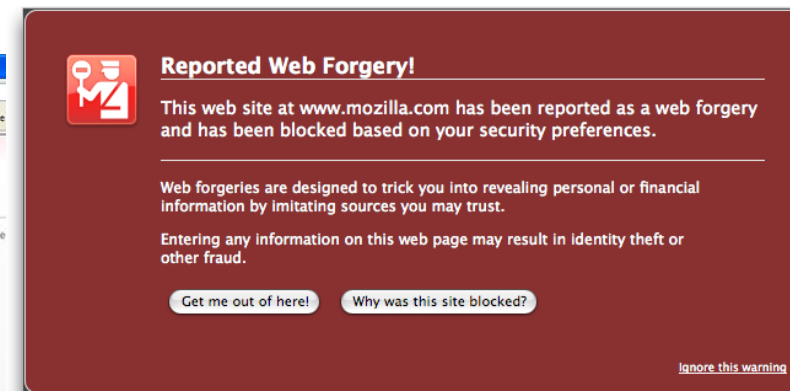
Active vs. Passive Warnings

- Active warnings significantly more effective
 - **Passive (IE): 100% clicked, 90% phished**
 - **Active (IE): 95% clicked, 45% phished**
 - **Active (Firefox): 100% clicked, 0% phished**



Passive (IE)

Active (IE)



Active (Firefox)

FYI: Site Authentication Image

Bank of America | Online Banking | SiteKey | Verify SiteKey - Windows Internet Explorer

https://sitekey.bankofamerica.com/sas/signonSetup.do

Bank of America | Online Banking | ...


Bank of America Higher Standards Online Banking

Confirm that your SiteKey is correct

If you recognize your SiteKey, you'll know for sure that you are at the valid Bank of America site. Confirming your SiteKey is also how you'll know that it's safe to enter your Passcode and click the **Sign In** button.

An asterisk (*) indicates a required field.

Your SiteKey:
pelicans



If you don't recognize your personalized SiteKey, don't enter your Passcode.

* Passcode:
(4 - 20 Characters, case sensitive)

Sign In


If you don't recognize your personalized "SiteKey", don't enter your Passcode


Modern anti-phishing

- Largely driven by Google Safe Browsing
 - Browser sends 32-bit prefix of hash(url)
 - API says: good or bad



Modern warnings

 **Dangerous** | testsafebrowsing.appspot.com/s/phishing.html



Deceptive site ahead

Attackers on **testsafebrowsing.appspot.com** may trick you into doing something dangerous like installing software or revealing your personal information (for example, passwords, phone numbers, or credit cards). [Learn more](#)

[Details](#) [Back to safety](#)



Deceptive site ahead

Firefox blocked this page because it may trick you into doing something dangerous like installing software or revealing personal information like passwords or credit cards.

Advisory provided by [Google Safe Browsing](#).

Go back

See details





The page ahead may try to charge you money

These charges could be one-time or recurring and may not be obvious.

Proceed

Go back





The site ahead contains malware

Attackers currently on **testsafebrowsing.appspot.com** might attempt to install dangerous programs on your computer that steal or delete your information (for example, photos, passwords, messages, and credit cards). [Learn more](#)

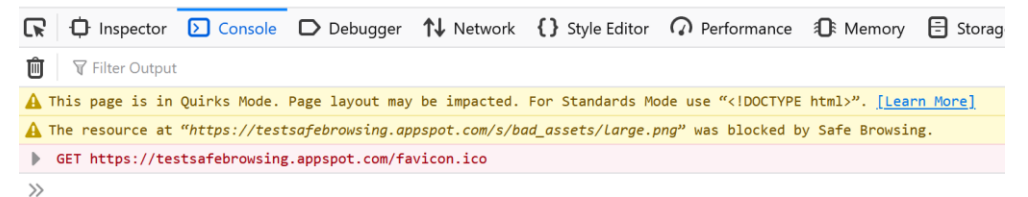
Details

Back to safety



5/19/2023

CSE 484 - Spring 2023



33

Which warning is 'better'?

- For user security?
- For user agency?
- For user understanding?
- For... what?

Side channels

Side-channels: conceptually

- A program's implementation (that is, the final compiled version) is different from the conceptual description
- Side-effects of the difference between the implementation and conception can reveal unexpected information
 - Thus: Side-channels

Detour: Covert-channels

- We'll see many unusual ways to have information flow from thing A to thing B
- If this is an *intentional* usage of side effects, it is a covert channel
- *Unintentional* means it is a side-channel
- The same *mechanism* can be used as a covert-channel, or abused as a side-channel

Side Channel Attacks

- Most commonly discussed in the context of cryptosystems
- But also prevalent in many contexts
 - E.g., we discussed the TENEX password implementation
 - E.g., we discussed browser fingerprinting

Why should we care about side-channels?

- Compromises happen via ‘simple’ methods
 - Phishing
 - Straight-forward attacks
- Embedded systems *do* see side-channel attacks
- “High Security” systems *do* see side-channel attacks



And they are getting more impactful...

- “The [Secret Network](#) has been vulnerable to the [xAPIC](#) and [MMIO vulnerabilities](#) that were publicly disclosed on August 9, 2022. These vulnerabilities could be used to extract the *consensus seed*, a master decryption key for the private transactions on the Secret Network. Exposure of the consensus seed would enable the complete retroactive disclosure of all Secret-4 private transactions since the chain began. We have helped Secret Network to deploy mitigations, especially the Registration Freeze on October 5, 2022.”



SGX.Fail

Timing Side-Channels

- Duration of a program (or operation) reveals information
- TENEX case



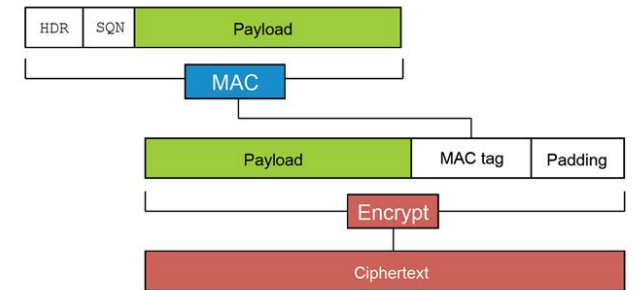
TENEX attack (for real)

- TENEX had an early *memory paging system*
- The original attack used page faults, not timing
 - Timing would've also worked 😊



Timing side-channels: round 2

- Cryptographic implementations fall down
 - #1 target for timing attacks
 - Extremely common to find vulnerabilities
- [“Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems”](#)
 - Was very far from the last paper on the topic



Attacking cryptography with side-channels

- ANY leakage is bad
 - E.g. 1 bit of key leaking is ‘catastrophic’
- Cryptographic implementations are complex
 - Many layers of protocols

Example Timing Attacks

- **RSA:** Leverage key-dependent timings of modular exponentiations
 - <https://www.rambus.com/timing-attacks-on-implementations-of-diffie-hellman-rsa-dss-and-other-systems/>
 - <http://crypto.stanford.edu/~dabo/papers/ssl-timing.pdf>
- **Block Ciphers:** Leverage key-dependent cache hits/misses

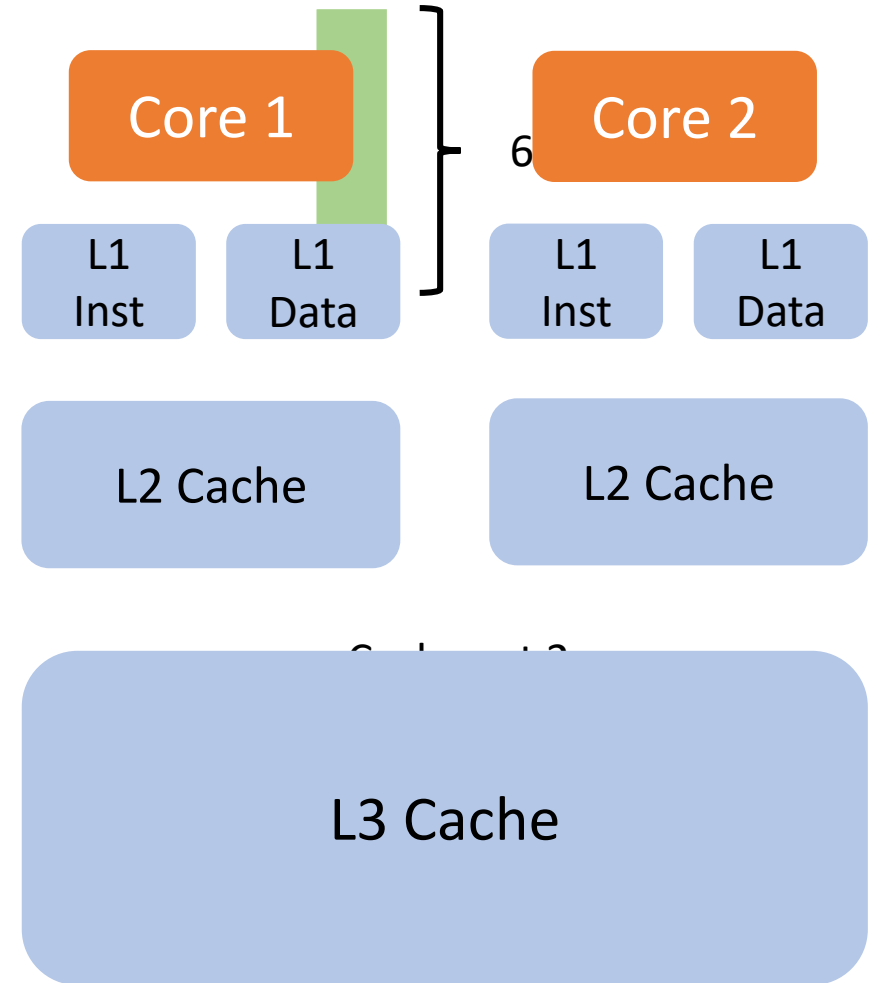
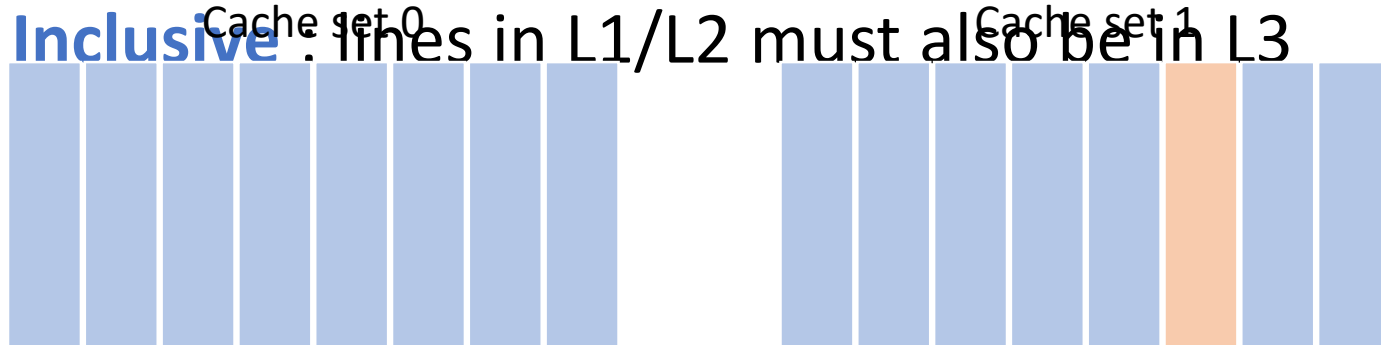
Cache side-channels

Cache side-channels

- **Idea:** The cache's current state implies something about prior memory accesses
- **Insight:** Prior memory accesses can tell you a lot about a program!

Cache Basics

- **Cache lines** : fixed-size units of data
- **Cache set** : holds multiple cache lines
- **Set index** : assigns cache line to cache set
- **Eviction** : removing cache lines to make room
- **L1, L2, L3** : different levels of cache
- **Inclusive** : lines in L1/L2 must also be in L3



Cache Attacks: Structure



Pre-Attack

Active Attack

Analysis

Many thanks to Craig Disselkoen for the animations.

Pre-Attack

Timing threshold
Eviction set

Active Attack

Prime targeted set

Wait

Active Attack

[Timed] Prime targeted set



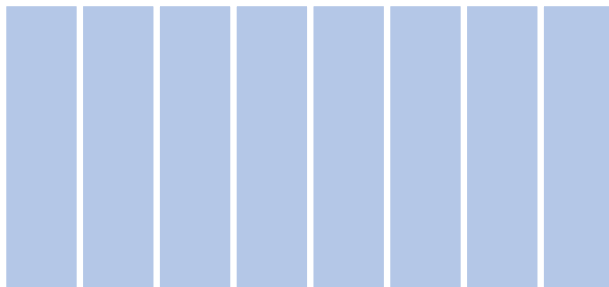
Victim accesses targeted set

Analysis

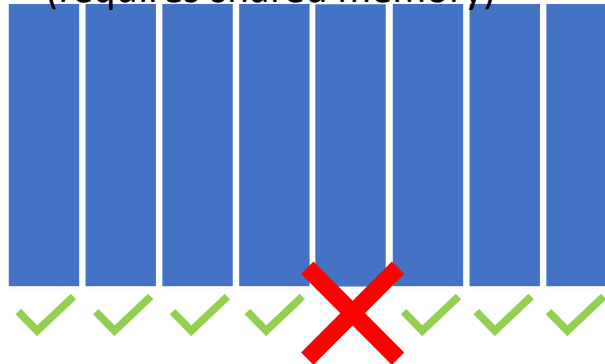
Victim access if
time > threshold

PRIME+PROBE FLUSH+RELOAD

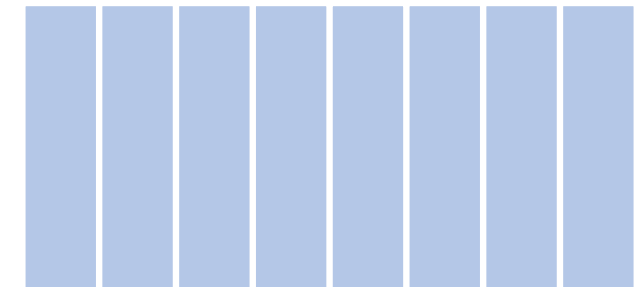
Cache set 0



Cache set 1
(requires shared memory)



Cache set 2



FLUSH + RELOAD

- Even simpler!
- Kick line L out of cache
- Let victim run
- Access L
 - Fast? Victim touched it
 - Slow? Victim didn't touch it

Cache attacks wrapup

- Cache attacks are a core element of many side-channels
- Generally “assumed to work” these days
- New variations/tricks/mitigations published constantly
- Randomized caches are the current hotness