CSE 484 / CSE M 584: Computer Security and Privacy

Usable Security

Spring 2016

Franziska (Franzi) Roesner
franzi@cs.washington.edu

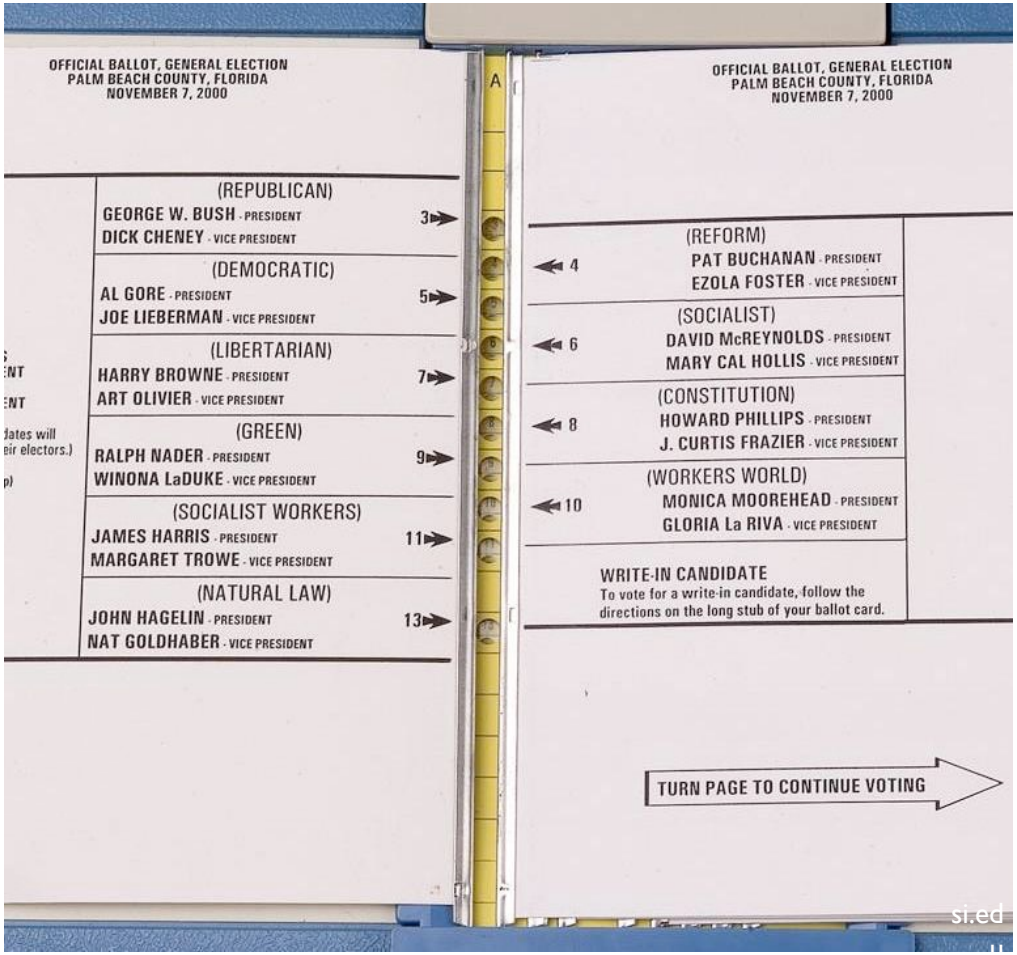
Thanks to Dan Boneh, Dieter Gollmann, Dan Halperin, Yoshi Kohno, John Manferdelli, John Mitchell, Vitaly Shmatikov, Bennet Yee, and many others for sample slides and materials ...

Looking Ahead

- It's almost the end of the quarter 
- Friday: lockpicking ☺
- Monday: holiday ☺

- **Homework #3 due Friday (5/27)**
- **Final Project Checkpoint #2 due Monday (5/31)**
- **Lab #3 due next Friday (6/3)**

Poor Usability Causes Problems



AP

Importance in Security

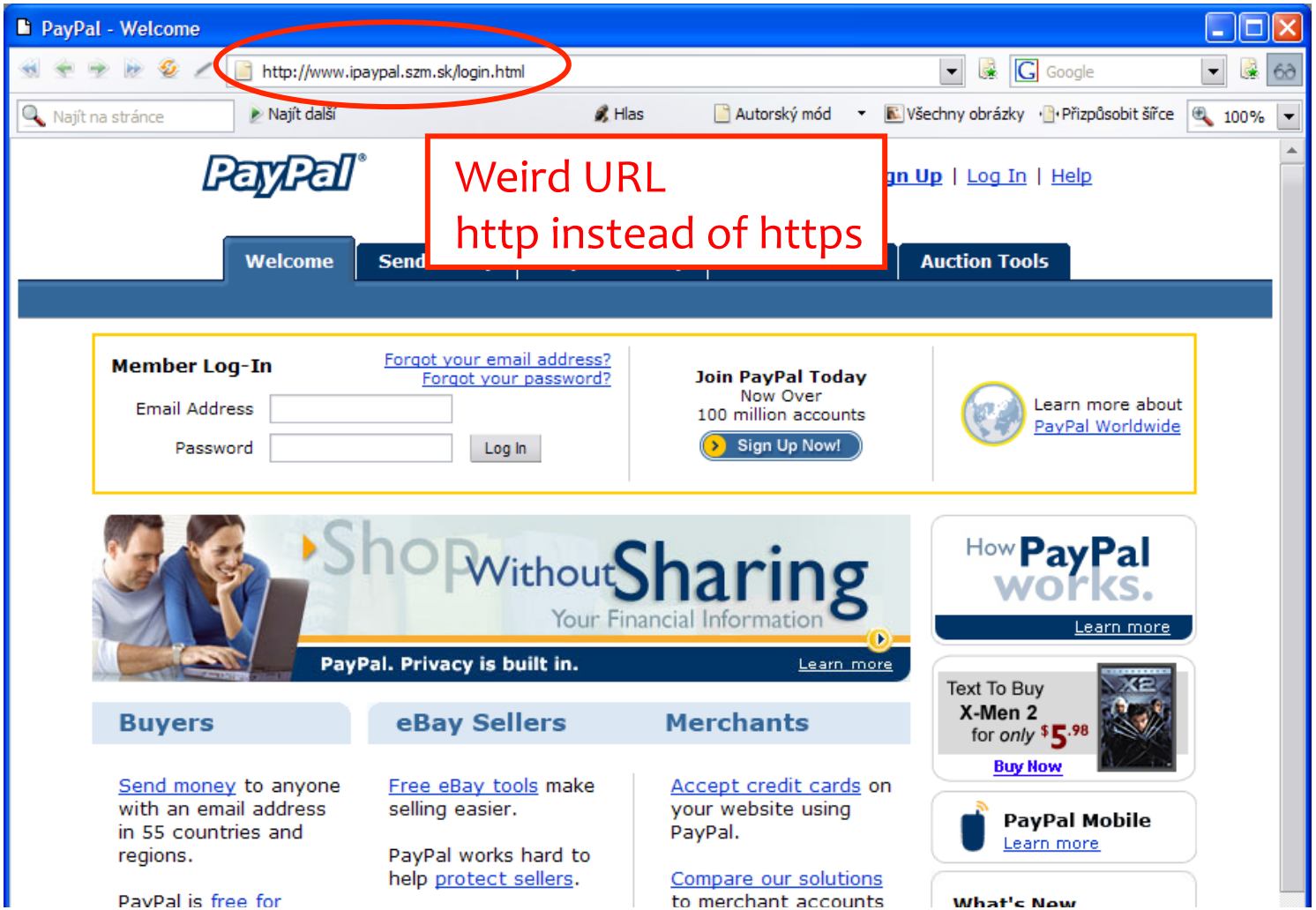
- Why is usability important?
  - People are the critical element of any computer system
    - People are the real reason computers exist in the first place
  - Even if it is **possible** for a system to protect against an adversary, people may use the system in other, **less secure** ways

Today

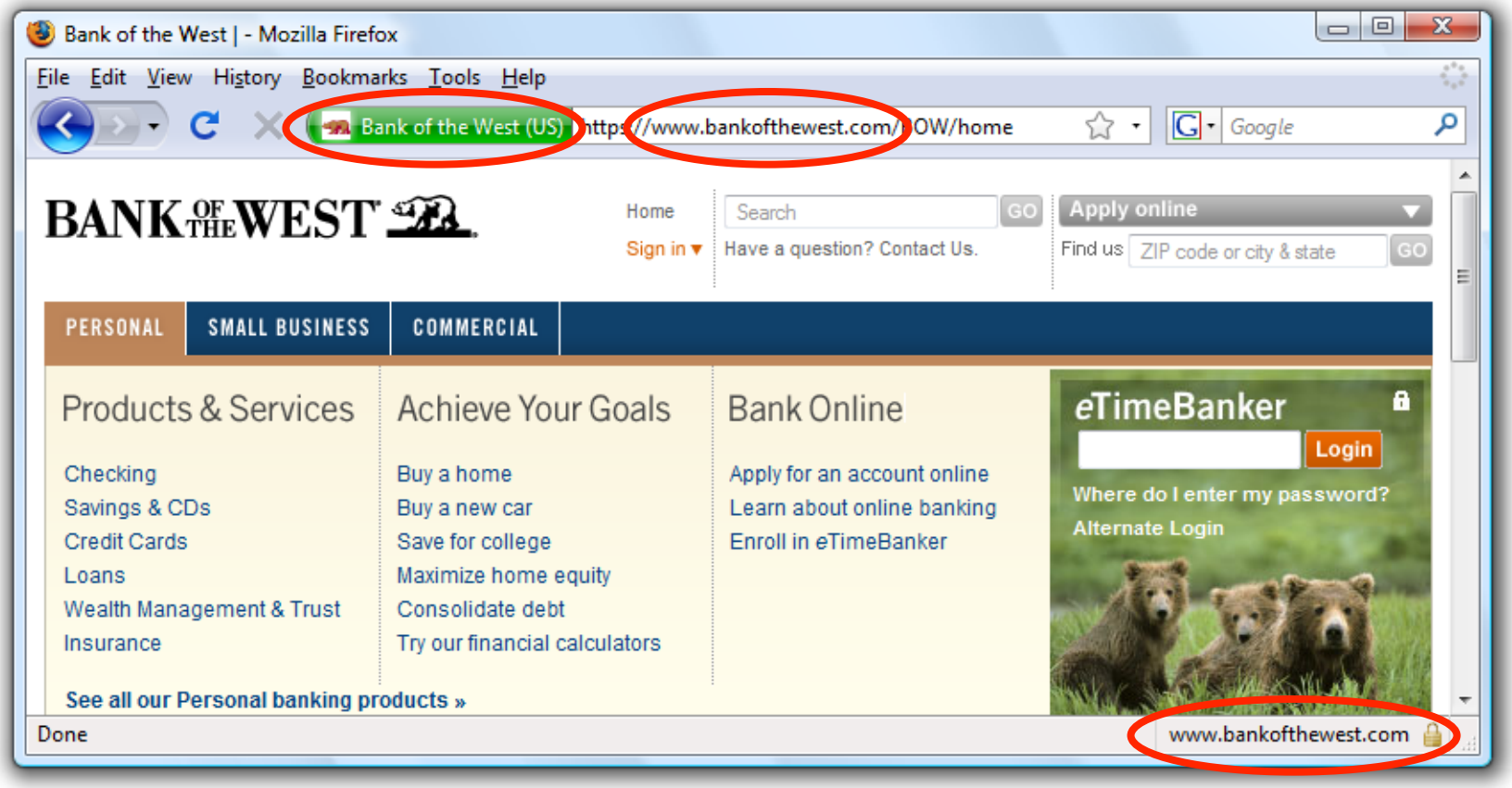
- 3 case studies
  - Phishing
  - SSL warnings
  - Password managers
- Step back: root causes of usability problems, and how to address

Case Study #1: Phishing

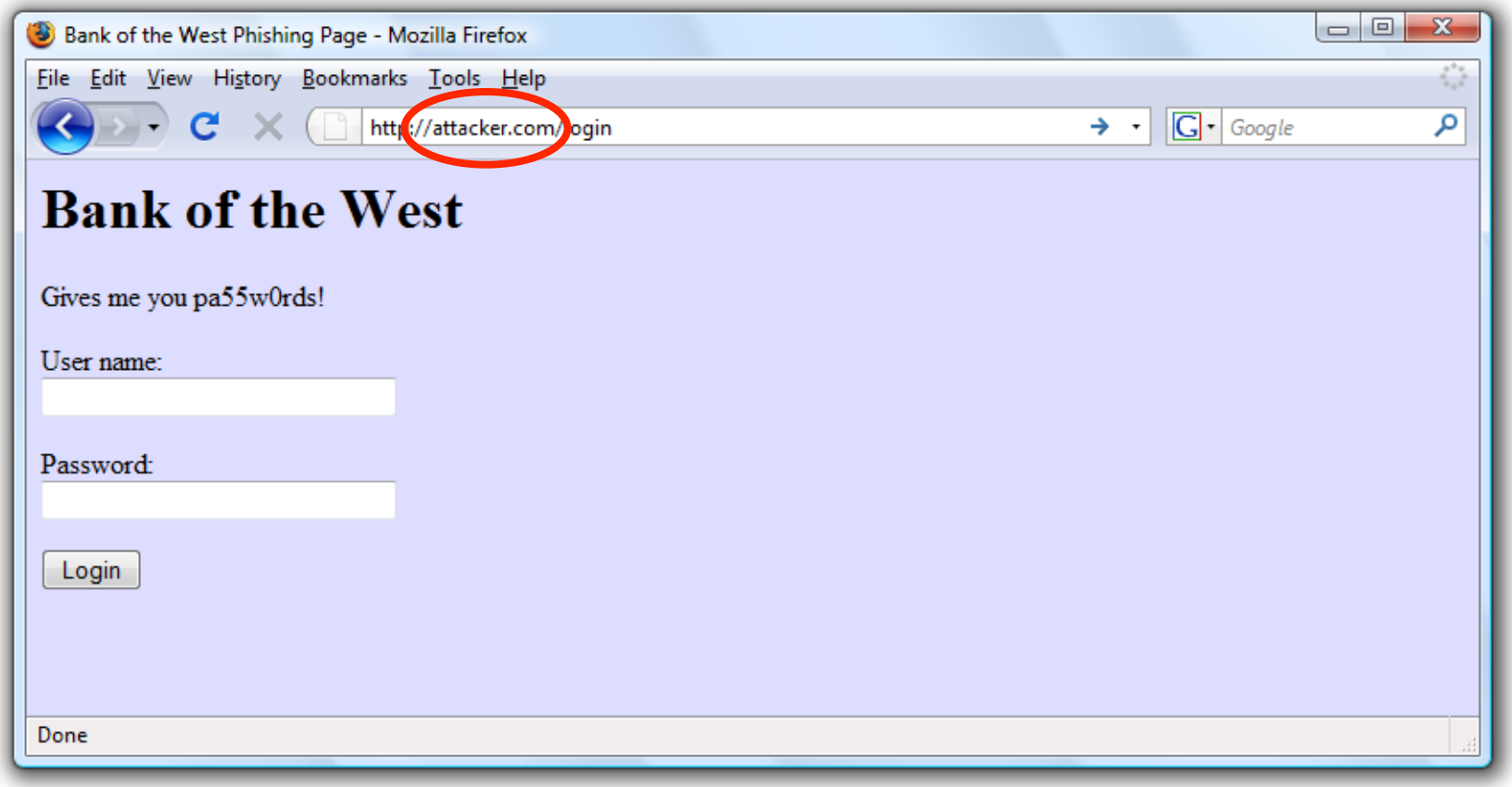
A Typical Phishing Page



Safe to Type Your Password?

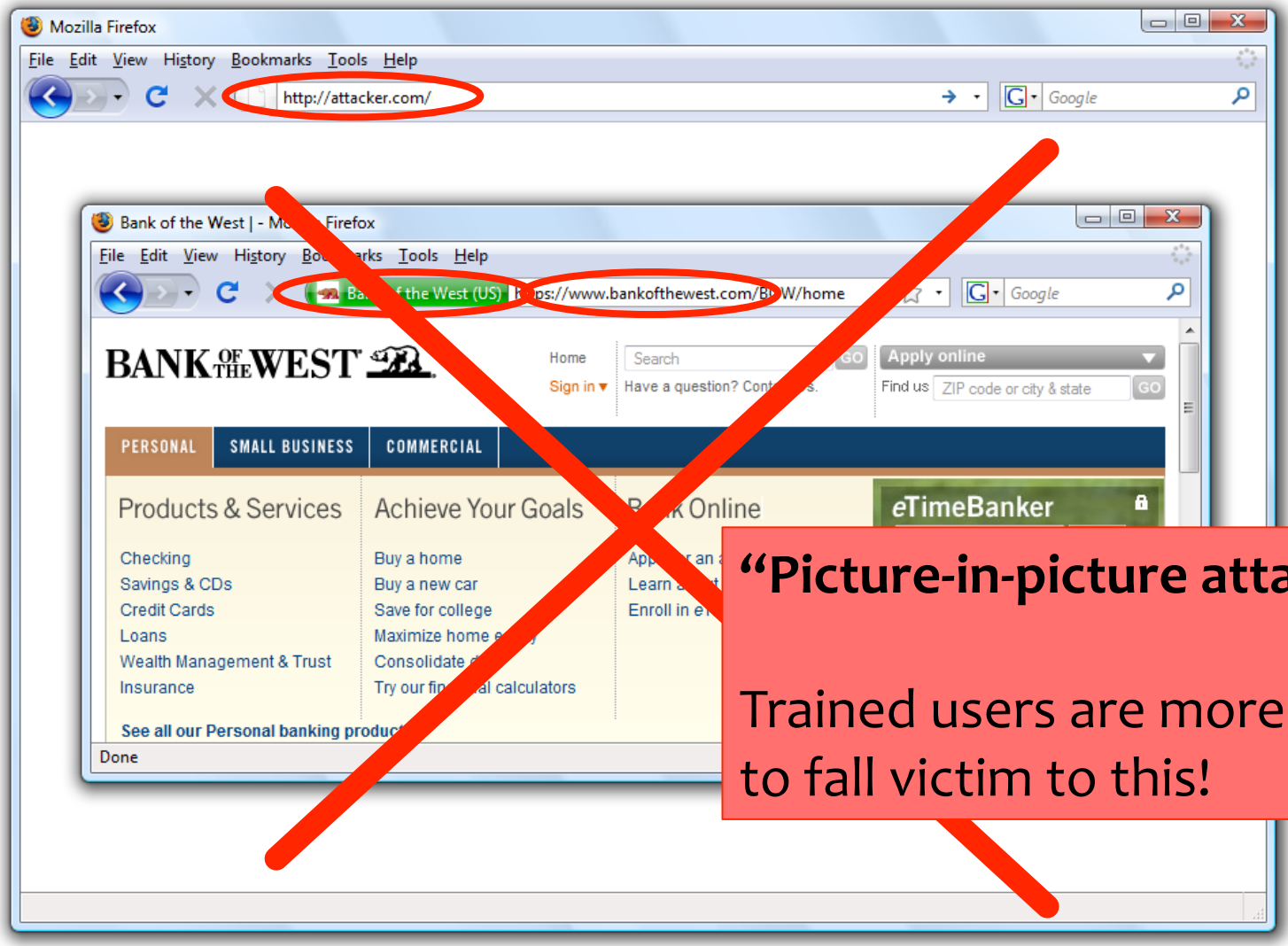


Safe to Type Your Password?



Safe to Type Your Password?

Safe to Type Your Password?



"Picture-in-picture attacks"

Trained users are more likely to fall victim to this!

Experiments at Indiana University

- Reconstructed the social network by crawling sites like Facebook, MySpace, LinkedIn and Friendster
- Sent 921 Indiana University students a spoofed email that appeared to come from their friend
- Email redirected to a spoofed site inviting the user to enter his/her secure university credentials
  - Domain name clearly distinct from indiana.edu
- 72% of students entered their real credentials into the spoofed site

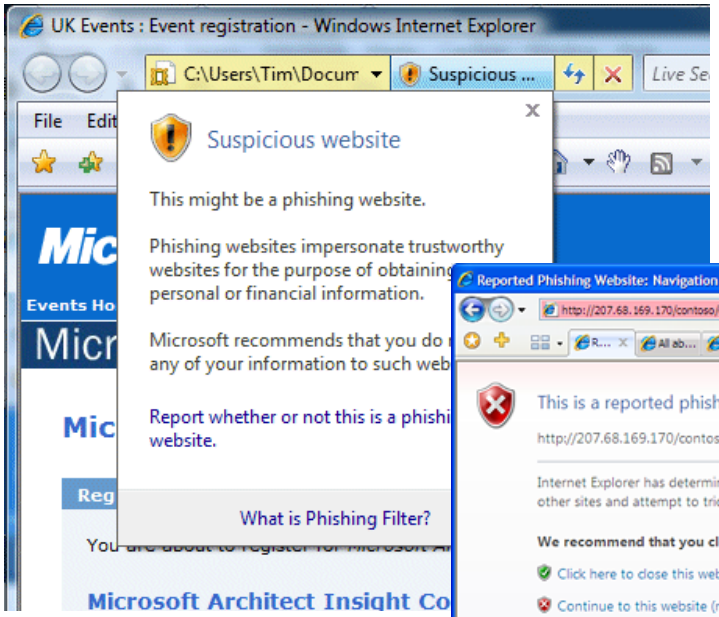
More Details

- Control group: 15 of 94 (16%) entered personal information
- Social group: 349 of 487 (72%) entered personal information

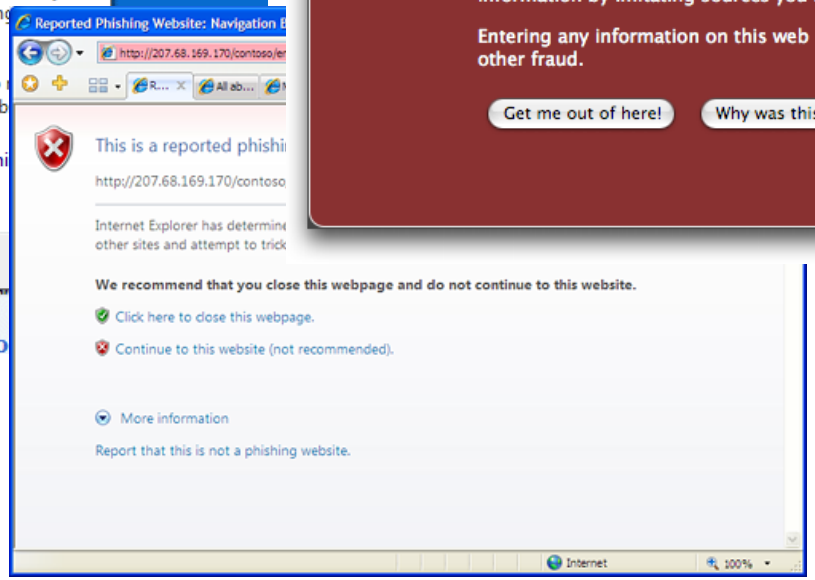
- 70% of responses within first 12 hours
- Adversary wins by gaining users' trust

- Also: If a site looks "professional", people likely to believe that it is legitimate

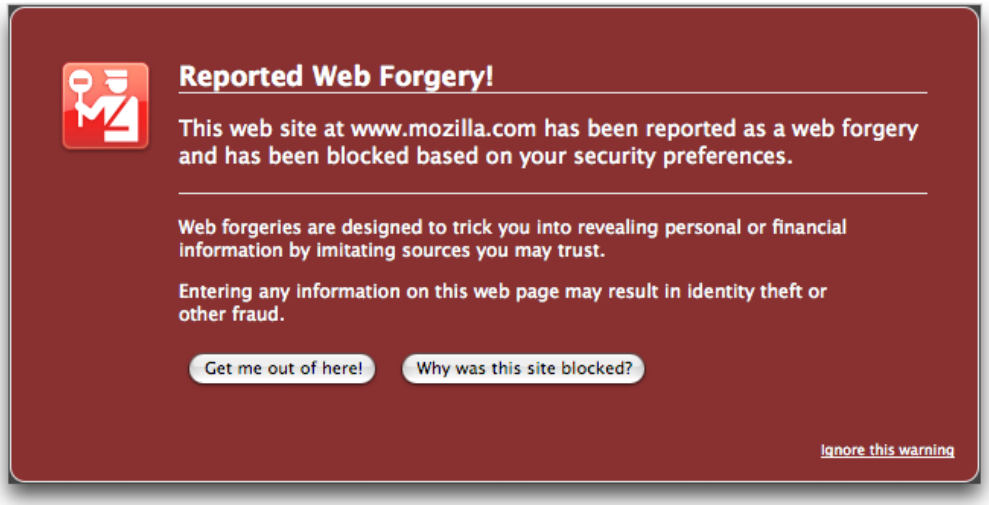
Phishing Warnings



Passive (IE)



Active (IE)



Active (Firefox)

[Egelman et al.]

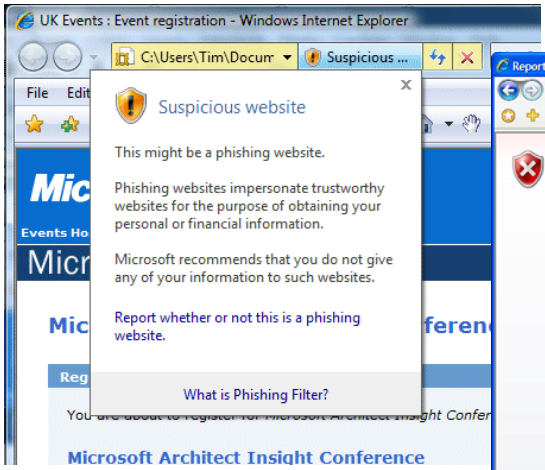
Are Phishing Warnings Effective?

- CMU study of 60 users
- Asked to make eBay and Amazon purchases
- All were sent phishing messages in addition to the real purchase confirmations
- Goal: compare <u>active</u> and <u>passive</u> warnings

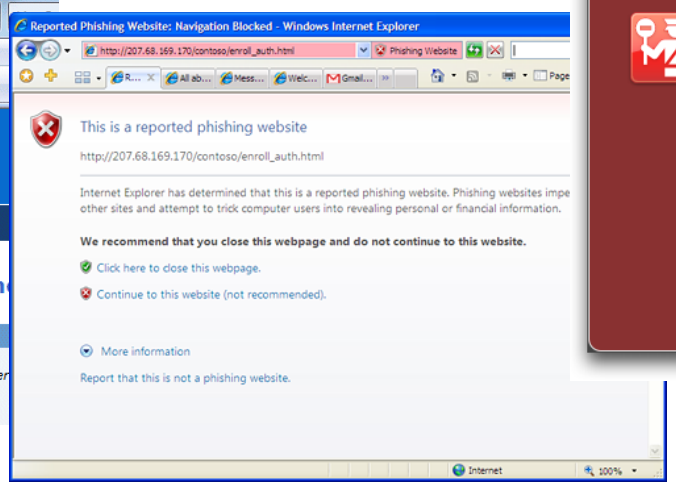
[Egelman et al.]

Active vs. Passive Warnings

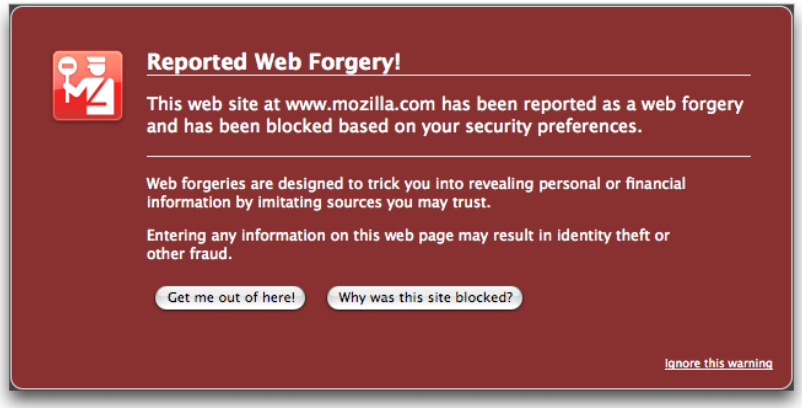
- Active warnings significantly more effective
  - Passive (IE): 100% clicked, 90% phished
  - Active (IE): 95% clicked, 45% phished
  - Active (Firefox): 100% clicked, 0% phished



Passive (IE)



Active (IE)



Active (Firefox)

[Egelman et al.]

User Response to Warnings

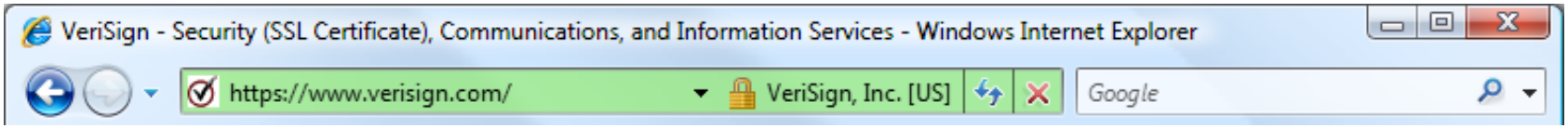
- Some fail to notice warnings entirely
  - Passive warning takes a couple of seconds to appear; if user starts typing, his keystrokes dismiss the warning
- Some saw the warning, closed the window, went back to email, clicked links again, were presented with the same warnings… repeated 4-5 times
  - Conclusion: "website is not working"
  - Users never bothered to read the warnings, but were still prevented from visiting the phishing site
  - Active warnings work!

[Egelman et al.]

Why Do Users Ignore Warnings?

- Don't trust the warning
  - "Since it gave me the option of still proceeding to the website, I figured it couldn't be that bad"
- Ignore warning because it's familiar (IE users)
  - "Oh, I always ignore those"
  - "Looked like warnings I see at work which I know to ignore"
  - "I thought that the warnings were some usual ones displayed by IE"
  - "My own PC constantly bombards me with similar messages"

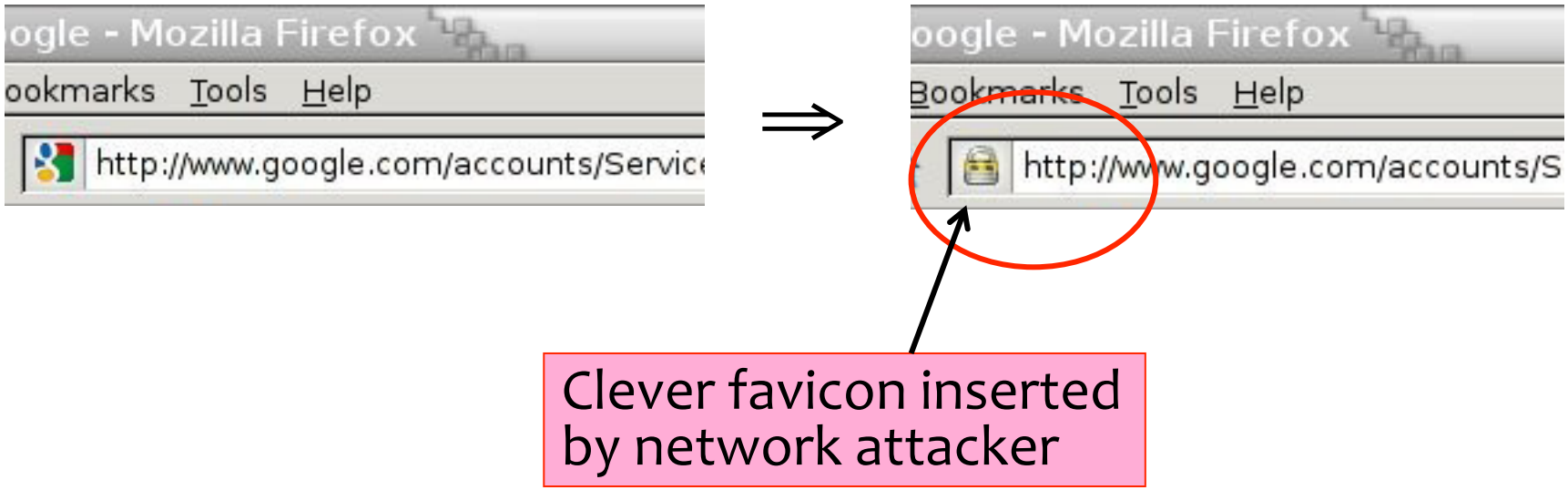
The Lock Icon



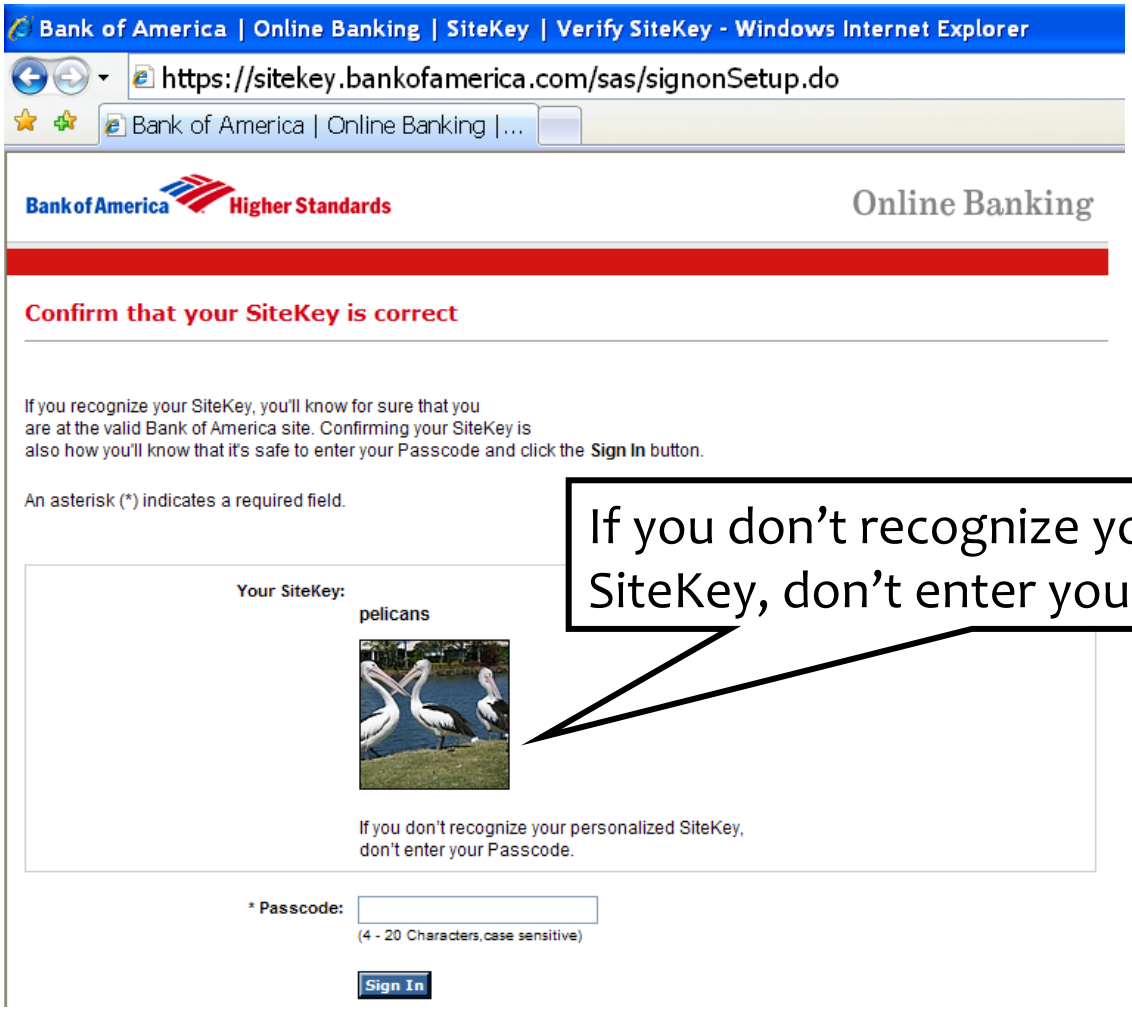
- Goal: identify secure connection
  - SSL/TLS is being used between client and server to protect against active network attacker
- Lock icon should only be shown when the page is secure against network attacker
  - Semantics subtle and not widely understood by users
  - Whose certificate is it??
  - Problem in user interface design

[Moxie Marlinspike]

Will You Notice?



Site Authentication Image (SiteKey)



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

Do These Indicators Help?

- "The Emperor's New Security Indicators"
  - http://www.usablesecurity.org/emperor/emperor.pdf

		Group				
Score	*First chose* not *to enter password...*	*1*	*2*	*3*	*1 ∪ 2*	*Total*
0	upon noticing HTTPS absent	0 0%	0 0%	0 0%	0 0%	0 0%
1	after site-authentication image removed	0 0%	0 0%	2 9%	0 0%	2 4%
2	after warning page	8 47%	5 29%	12 55%	13 37%	25 44%
3	never (always logged in)	10 53%	12 71%	8 36%	22 63%	30 53%
	Total	18	17	22	35	57

Users don't notice the **absence** of indicators!

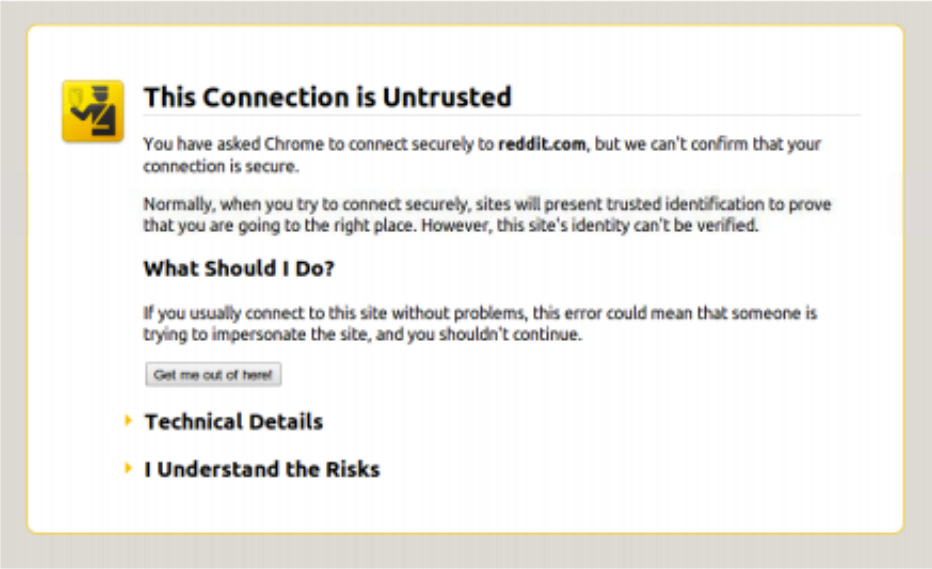
Case Study #2: Browser SSL Warnings

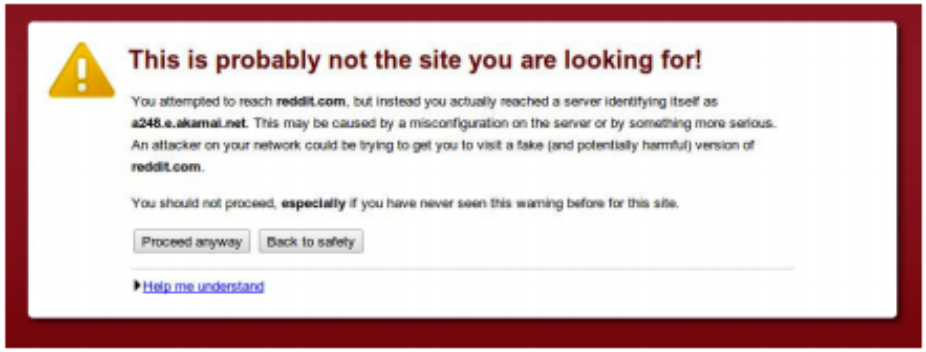
- Design question: How to alert the user if a site's SSL certificate is untrusted?

[Felt et al.]

Firefox vs. Chrome Warning

33% vs. 70% clickthrough rate





[Felt et al.]

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.

[Felt et al.]

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.

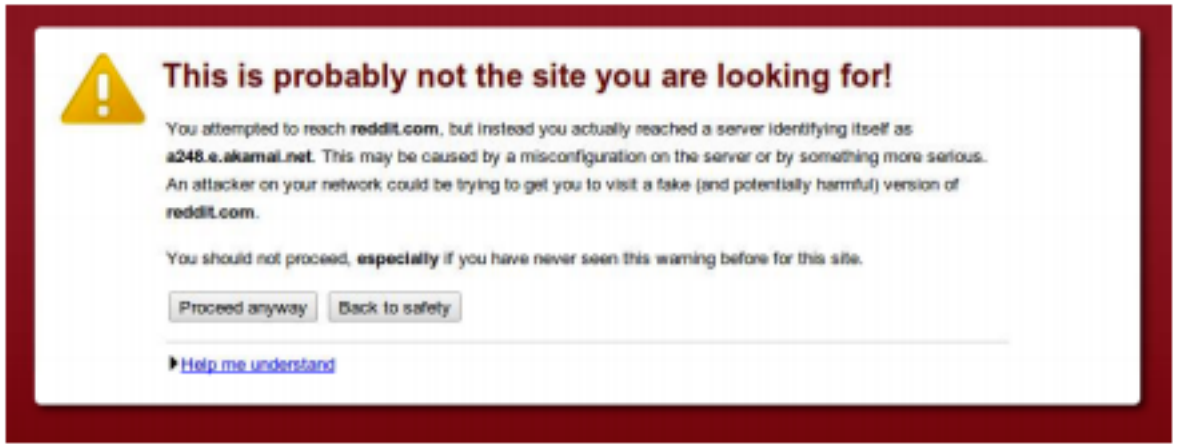


Figure 1. The default Chrome SSL warning (Condition 1).

[Felt et al.]

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

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

[Felt et al.]

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.

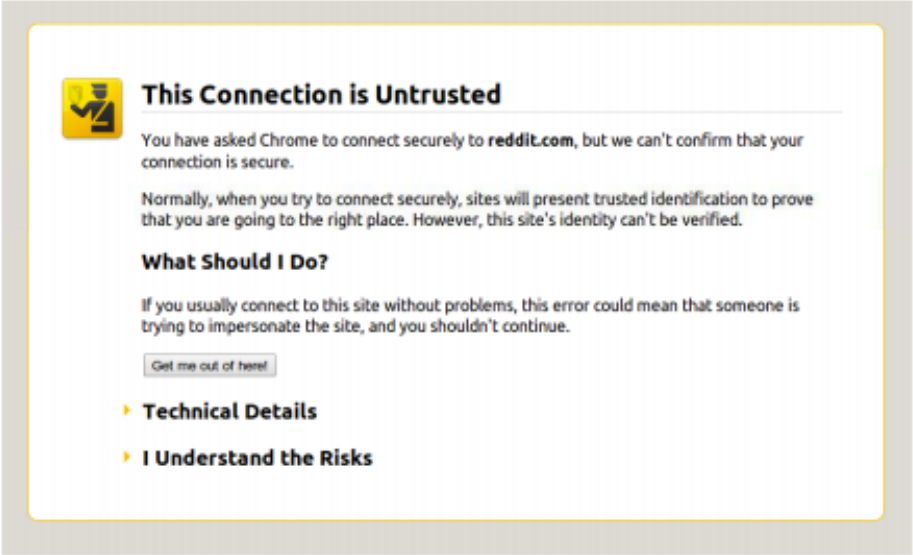


Figure 2. The mock Firefox SSL warning (Condition 5).

[Felt et al.]

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.

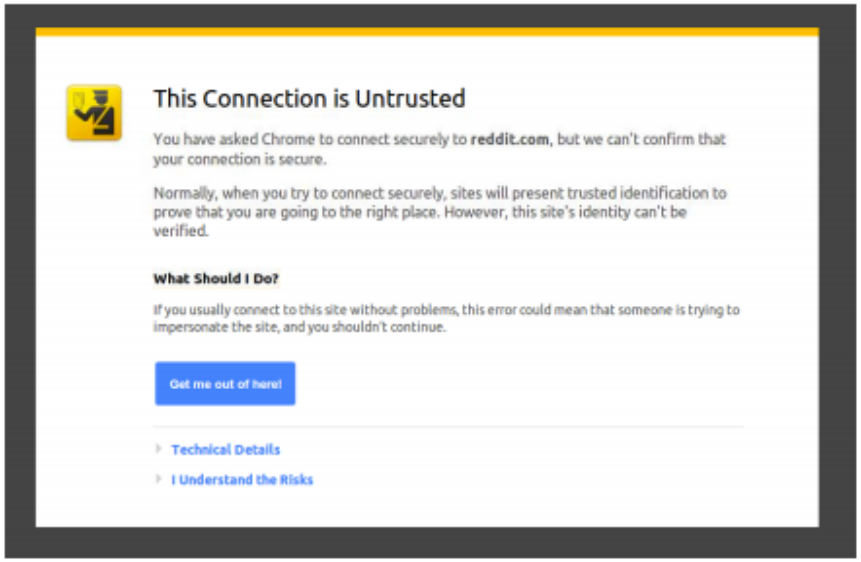
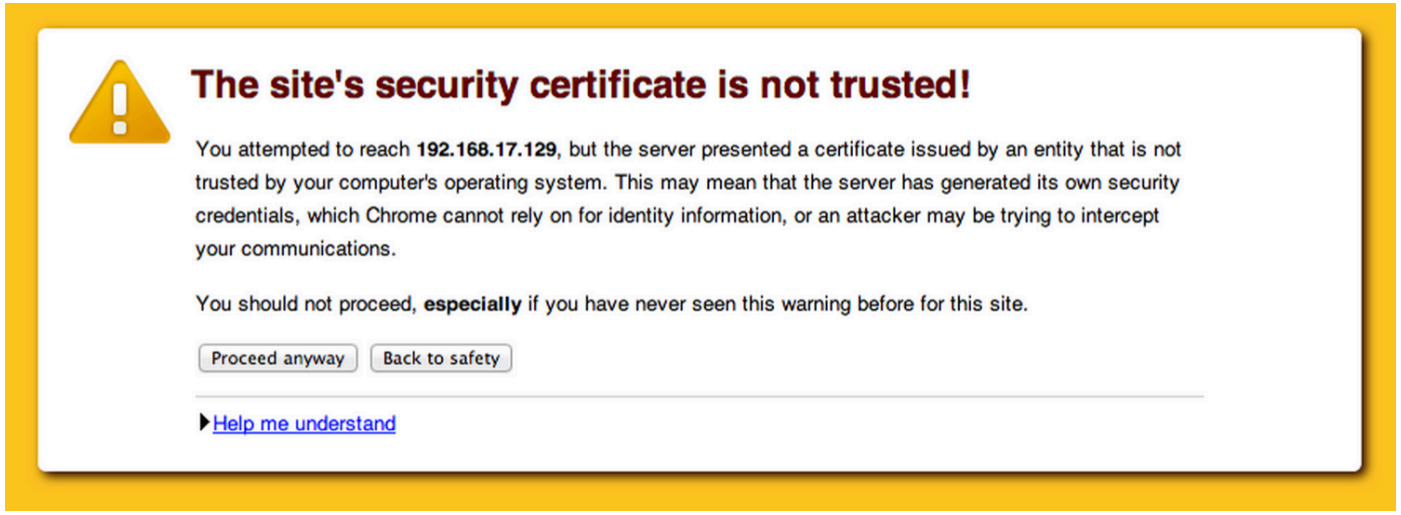


Figure 3. The Firefox SSL warning with Google styling (Condition 7).

[Felt et al.]

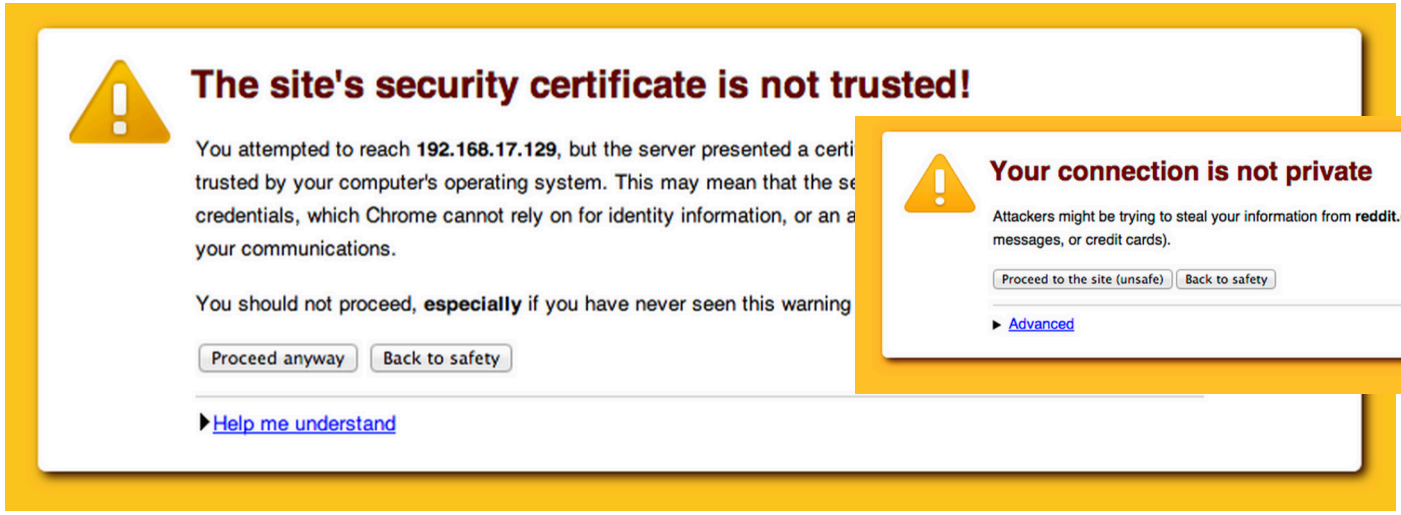
Opinionated Design Helps!

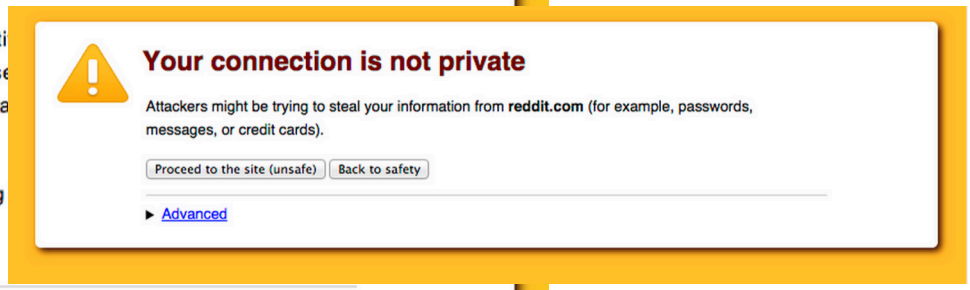


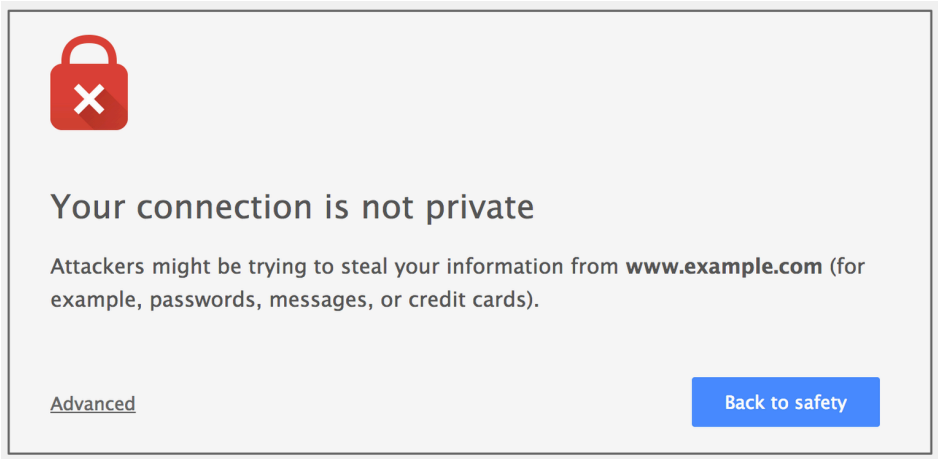
Adherence	N
30.9%	4,551

[Felt et al.]

Opinionated Design Helps!



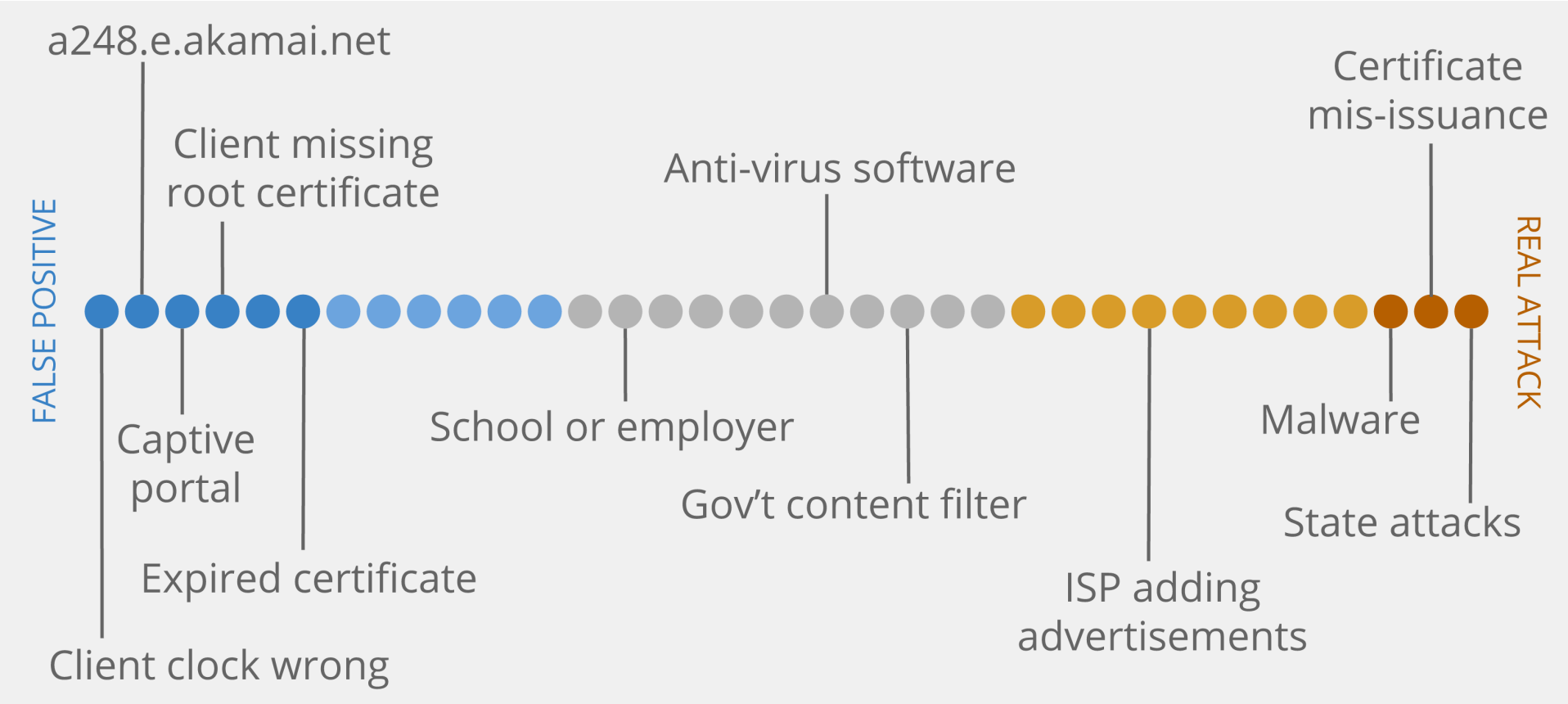




Adherence	N
30.9%	4,551
32.1%	4,075
58.3%	4,644

[Felt et al.]

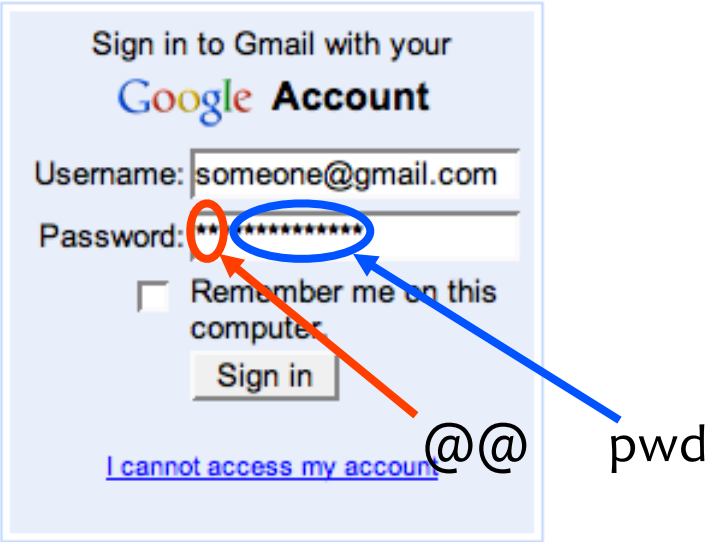
Challenge: Meaningful Warnings



Case Study #3: Password Managers

- Password managers handle creating and "remembering" strong passwords
- Potentially:
  - **Easier** for users
  - More **secure**
- Examples:
  - PwdHash (Usenix Security 2005)
  - Password Multiplier (WWW 2005)

PwdHash



@@ in front of passwords to protect; or F2

sitePwd = Hash(pwd,domain)

Prevent phishing attacks

Password Multiplier

Activate with Alt-P or double-click

sitePwd = Hash(username, pwd, domain)

Both solutions target simplicity and transparency.

[Chiasson, van Oorschot, Biddle]

Usability Testing

- Are these programs usable? If not, what are the problems?
- Two main approaches for evaluating usability:
  - Usability inspection (no users)
    - Cognitive walkthroughs
    - Heuristic evaluation
  - User study
    - Controlled experiments
    - Real usage

[Chiasson, van Oorschot, Biddle]

Task Completion Results

	Success	Potentially Causing Security Exposures			
		Dangerous Success	Failures		
			Failure	False Completion	Failed due to Previous
PwdHash					
Log In	48%	44%	8%	0%	N/A
Migrate Pwd	42%	35%	11%	11%	N/A
Remote Login	27%	42%	31%	0%	N/A
Update Pwd	19%	65%	8%	8%	N/A
Second Login	52%	28%	4%	0%	16%
Password Multiplier					
Log In	48%	44%	8%	0%	N/A
Migrate Pwd	16%	32%	28%	20%	N/A
Remote Login	N/A	N/A	N/A	N/A	N/A
Update Pwd	16%	4%	44%	28%	N/A
Second Login	16%	4%	16%	0%	16%

[Chiasson, van Oorschot, Biddle]

Problem: Transparency

- Unclear to users whether actions successful or not.
  - Should be obvious when plugin activated.
  - Should be obvious when password protected.
- Users feel that they should be able to know their own password.

[Chiasson, van Oorschot, Biddle]

Problem: Mental Model

- Users seemed to have misaligned mental models
  - Not understand that one needs to put "@@" before *each* password to be protected.
  - Think different passwords generated for each session.
  - Think successful when were not.
  - Not know to click in field before Alt-P.
  - Don't understand what's happening: "Really, I don't see how my password is safer because of two @'s in front"

[Chiasson, van Oorschot, Biddle]

When "Nothing Works"

- Tendency to try all passwords
  - A poor security choice – phishing site could collect many passwords!
  - May make the use of PwdHash or Password Multiplier *worse* than not using any password manager.
- Usability problem leads to security vulnerabilities.
  - Theme in course: sometimes things designed to increase security can also increase other risks

Question

- **Q.** What are the root causes of usability issues in computer security?

Question

- **Q.** What approaches can we take to mitigate usability issues in computer security?