eSWAT: A spyware-resistant virtual keyboard

William Allen Ph.D., Dr. Richard Ford D.Phil., Aldwin Saugere, Florida Institute of Technology rford@fit.edu



What are we talking about?

- Why we've started seeing virtual keyboards online
- How many current virtual keyboards are not ideal
- What can be done using simple AJAX techniques
- Why this matters
- Also: Demo of eSWAT in action

Authentication

- For the consumer level, primarily *reusable* credentials (username, password, maybe a security question)
- Usually, trivially sniffable by any attacker
- Need to balance sense of security with usability with actual security...

Keystroke logging

- Ongoing problem, and certainly not new
- Happens in the physical keyboard as well as using software
- Difficult to detect generically

Secure Data Entry

- Keyloggers and botnets continue to be found
- More and more information is accessible online
- Stealing someone's account is actually pretty useful
- Access credentials are a problem at the home user/remote worker level

Other Solutions: Smart Cards et al.

- Of course this is the *right* way to do it
- Provable security
- No reliance on security through obscurity
- "Something you have, something you know"

Cost and ROI

- Not for the defender, but for the attacker!
- The more effort required, the less financial motivation, as we tip the cost/return calculations
- Very similar arguments can be made for all Financially-motivated malware

Microsoft[™] Virtual Keyboard



Citibank

Welcome to Citibank Online

Use your <u>keyboard</u> to enter your 16 digit ATM / Debit Card Number or Credit Card Number or Loan ID or World Money Card

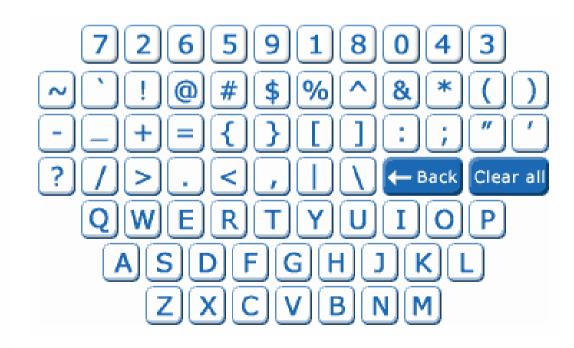
Save my card number and create username

Enter your Password using the Mouse

⊙ IPIN © QPIN

Internet Password is not case sensitive.





Our Goals

- Can we do better?
 - Yes, *much* better
- Demonstrate how much more can be done using already-extant technologies
 - AJAX and Web 2.0 provide everything we need...
- People are starting to use these systems... what's the exposure and how much do they help?
- If we're going to *feel* more secure, let's see what we can put behind it!

Our Requirements

- No installable component on the machine
- No special hardware
- Provides some level of protection that is *concrete*

eSWAT



Modes of Operation

- 1. Offset mouse, key rollover
- 2. No key rollover, cursor vanishes on mouse down
- 3. Cursor offset randomized
- 4. No mouse clicks system based on time over key
- 5. As (4), but with keyboard movement
- 6. Mouse clicks only count at certain times

Demonstration

- This slide intentionally left blank $\textcircled{\odot}$

Attacks on eSWAT

- We'll take a look at:
 - 1. Keystroke logging
 - 2. Screen capture
 - 3. Network interception
 - 4. Dynamic disassembly
 - 5. Replacing (substituting) eSWAT

Keystroke loggers

• Fail!

No keyboard input

Screen capture

- Fails mostly...
 - In higher security mode, you need continuous logging
 - Not automatable (hard for a computer to parse)
 - Rather memory intensive (need to capture at fairly high resolution)

Network Interception

- Fails!
 - Yes, an attacker can see inside the SSL connection...
 - But why not use a one time pad, "baked in" to the download?
 - Thus, seeing inside SSL doesn't help!

Demonstration

• Using eSWAT with our own website...

Dynamic Disassembly

- Succeeds but...
 - It's really difficult
 - Use polymorphism/metamorphism
 - Use code obfuscation
 - The information is in there but probably requires manual recovery

Demonstration

• Showing how eSWAT code is different upon each load

Replacing eSWAT

- Fails (mostly)...
 - Relies on users noticing the problem
 - Although the user has given up their password...
 - The "real" eSWAT can be customized for user look and feel
 - The trojan can't "pass through" the real login information (baked in pad)
 - The website can show how many failed logins

Weaknesses

- Shoulder surfing
 - Could use custom hardware, but if this is an option, there are better solutions

Session hijacking

- Although we have lost the session, we didn't give up the password!
- AJAX hooking
 - Fairly difficult to determine the results might be automatable

Future Work

- Easy for us to add new functionality
 - Anti-phishing (eSWAT knows where it is submitting the data from the form)
 - Easy to put in trivial changes (like randomizing the keyboard layout)
 - Not really that much more to be done!

Conclusions

- The current "state of the art" in virtual keyboards is fairly poor
- Virtual keyboards provide a cost-effective way of reducing risk for vendors
- New technologies provide significant benefits to buy down user exposure
- Remember: a product/approach doesn't have to be 100% secure to be useful!