Dr. Evil's Guide to Web 2.0

Rafal Los

Twitter: @RafalLos

Blog: http://www.hp.com/go/white-rabbit

Talk 4-1-1

Disclaimer:

For the duration of this talk and any surrounding conversations – thoughts, research and comments are my own and not that of my employer, friends, family or anyone else I know unless explicitly stated.

Warning:

This talk is heavily "show-n-tell" based, so if you're reading the slides you probably will miss 90% of the content. Most of the things presented here are *real exploits* and should be taken carefully – so don't go hacking into some company and claiming I told you to do it.

Clear?

The Ground Rules

- ✓ Play along on your laptop! (*there are prizes*)
- ✓ Ask questions
- ✓ Participate, share, discuss
- ✓ Tweet this?…
  HashTag → #THOTCON

Let's do this…

MANDATORY BACKGROUND

OMG WTF is Web 2.0?

Web 2.0 is a bunch of shiny red lipstick on an old pig, with some modernizations

Web 2.0 is all about the **user experience**…

Web 2.0 is all about **rich content**…

Web 2.0 is all about **browser optimization**…

<script>alert('Part 1')</script>

BASIC COMPONENTS

I built a castle…

Crotchety Old Web

- ✓ HTML v1.0
- ✓ Synchronous
- ✓ "web pages"
- ✓ Simple txt editor
- ✓ HTML + JavaScript
- ✓ Browser renders HTML

Web 2.0 Sexy

- ✓ HTML v5.0
- ✓ Asynchronous
- ✓ "Web applications"
- ✓ 4 DVDs of Visual Studio
- ✓ HTML + AJAX, JavaScript, Flash, Silverlight …
- ✓ Plug-ins galore

What to Exploit?

So many broken parts, so little time

- Exploit trust
  - Between components
  - Between applications
  - Between sites
- Exploit bad code
  - Code with bugs
  - Code with TMI
- Exploit interoperability
  - Data exchange
- Exploit the user
  - Users are the weak link
- Exploit the browser?
- Many other options…

<script>alert('Part 2')</script>

SO MANY TARGETS

Client-side Objects

Why

- Profit?
- Free stuff?
- It's easy?
- Hard to get caught?

How

- Analysis tools
  - Client-side *decompiler**
  - Proxy
  - Text editor
  - Hex editor
- Your brain
- Patience

Social Media

Why

- Social media exposed APIs are ripe for exploitation
- Profit from people
- People are sheep (trust)
- Click-happy end users

How

- Exploit trust issues in social interactions
- RTF [api]M – plug-ins for social platforms
- Social engineering
- Legal (*but shady*) use of legitimate platforms

HTML v5 Hotness

Why

- HTMLv5 is a massive standard
- Most developers haven't read >25%
- So many cool new toys to play with

How

- Legal code → malicious purpose (ClickJacking?!)
- Stuff XSS into EXIF tags, used with the FileAPI operators in HTMLv5 (Photoshop online)
- Asynchronous application logic exploitation

<script>alert('Part 3')</script>

HAVING FUN

Game #1

Goal: Win Stuff

How do you "win" online (Flash) games or contests without all the hassle of playing?

- Identify a game
- "Open it up"
- Find the logic

- String constructor (+3)
- Show complete POST/ GET to "win" (+5)

Game #2

Goal: Bypass the Login

Who writes authentication in Flash?! Better still …who writes client-side authN?

- Find a login mechanism
- "Open it up"
- Identify authN logic

- Hidden URLs (+3)
- Bypass login (+5)

Game #3

Goal: Pwn a database

Believe it or not, developers have written client-side interfaces for a database. What could possibly go wrong?

- Find a database access point
- "Open it up"
- Identify DBConn str (+3)
- Connect to DB (+10)

How did you do that!?

Tools used

- Google search (+dorks)
- Hex Workshop
- HP SWFScan
  - Flash de-compiler
- Burp Suite
- Exiv2
- Notepad++
- wget (win32)