Introduction

- Browser ecosystem is at the forefront of the war
- How is the browser ecosystem adapting?
Overview

1. Browser ecosystem threat overview
2. Past and recent developments
3. Plug-ins detour
4. Looking to the future / malware trends
5. Blacklists as a defense-in-depth measure
6. New attack areas exposed by browsers
Browser ecosystem

System / network libraries

Plugins...

- Adobe
- Gears
- Flash Player
- PDF
- Music
- Video
Browser ecosystem: threats

1. Arbitrary code execution
   - Domain-isolated
   - Sandboxed
   - Unsandboxed
2. Cross-origin data theft
3. Web-app based leaks
Recent changes

- Increased file download security
  - E-mail clients
  - Warning dialogs
  - Protected mode execution
  - Admin controls
  - Anti-virus enhancements
  - Whitelist-based security
Recent changes

- Sandboxing in browsers
  - IE7 on Vista: protected mode sandbox
  - Chromium on XP: filesystem sandbox
  - Chromium on Vista: filesystem + protected mode sandbox
  - Chromium on Linux: chroot() sandbox
  - Chromium on Mac: seatbelt sandbox
Recent changes

- Sandboxing in browsers
Recent changes

- Auto-updating users
  - Now widely accepted as required for security
  - On board: Windows, Google Chrome, Firefox, ...
  - Recent: Apr 2010, Adobe Reader auto-updater out of beta
- Interesting auto-update paper
  - [http://www.techzoom.net/publications/silent-updates/](http://www.techzoom.net/publications/silent-updates/)
Auto-update
Recent changes

- Attacker focus on plug-ins
- Plug-in stats (Google Chrome v4.1):
  - 97%: Shockwave Flash
  - 86%: Adobe Acrobat
  - 66%: Java(TM) Platform SE 6
    - Only 14% fully up to date
  - 53%: Windows Media Player
  - 49%: Silverlight Plug-In
  - 39%: QuickTime Plug-In
Recent changes

● Attacker focus on plug-ins
● SANS Top Cyber Security Risks 2009
  "Priority One... vulnerabilities in commonly used programs such as Adobe PDF Reader, QuickTime, Adobe Flash and Microsoft Office"
● News articles on theregister.co.uk
  o June 2010, "Adobe lines up emergency Flash fix"
  o April 2010, "Java code-execution vuln exploited in drive-by attack"
  o April 2010, "PDF security hole opens can of worms"
  o July 2009, "New attacks exploit vuln in (fully-patched) Adobe Flash"
Plug-in detour

**Plug-ins**

**Google Gears 0.5.33.0** - Version: 0.5.33.0 (Disabled)
Description: These are the Gears that power the tubes! -)
Location: C:\Documents and Settings\evans\Local Settings\Application Data\Google\Chrome\Application\5.0.375.70\gears.dll
Enable

**Adobe Acrobat** - Version: 9.3.2.163 (Disabled)
Description: Adobe PDF Plug-In For Firefox and Netscape *9.3.2*
Location: C:\Program Files\Adobe\Reader 9.0\Reader\Browser\npdf32.dll
Enable

**Java Deployment Toolkit 6.0.180.7** - Version: 6.0.180.7 (Disabled)
Description: NPRuntime Script Plug-In Library for Java(TM) Deploy
Location: C:\Program Files\Java\jre6\bin\new_plugin\npdeploytk.dll
Enable

**Java(TM) Platform SE 6 U18** - Version: 6.0.180.7 (Disabled)
Description: Next Generation Java Plug-In 1.6.0_18 for Mozilla browsers
Location: C:\Program Files\Java\jre6\bin\new_plugin\npjp2.dll
Enable

**Oracle Jinitiator** - Version: 1.3, 1.29 (Disabled)
Description: Jinitiator 1.3.1.29 for Netscape Navigator (DLL Helper)
Location: C:\Program Files\Mozilla Firefox\plugins\npJinit13128.dll
Enable

**Microsoft® DRM** - Version: 9.00.00.4503 (Disabled)
Description: DRM Netscape Network Object
Location: C:\Program Files\Windows Media Player\npdrm2.dll
Enable

**Windows Media Player Plug-in Dynamic Link Library** - Version: 3.0, 3.02.629 (Disabled)
Description: npdeploy.dll
Plug-in detour
Plug-in detour
Future: sandboxing

- Safari
  - April 2010: WebKit2
    "WebKit2 is designed from the ground up to support a split process model..."

- Firefox
  - April 2010: Firefox 3.6.4 dev release
    ■ Plug-ins in separate process
  - July 2009: "Electrolysis" announced
    ■ Security as a long-term goal

- Plug-ins
  - Hard!
  - Browser as O/S specific
Future: plug-ins

- As browsers get more secure, less tolerance for poor plug-in security
- Internet Explorer: warns upon leaving protected mode
- Firefox: warns on out-of-date plug-ins
- Chromium: plug-ins inside auto-update umbrella; sandboxed PDF viewer
Future: soft spots

● Java plug-in
  ○ Very powerful => hard to sandbox
  ○ High potential for reliable exploits
    ■ April 2010, Ormandy: command-line error
    ■ May 2009 / April 2010, Koivu & Tinnes: deserialization bugs

● Operating system kernels
● Extension systems
Future: soft spots

- **Operating system kernels**
  - Big attack surface to escape sandboxes
  - Linux: ~300 syscalls
  - Mac: ??
  - Windows: ~1400 more complicated syscalls
  - Under-researched area (except Linux)

- **Kernel bug samples**
  - Jan 2010, Ormandy; Windows #GP Trap handler
  - Aug 2009, Tinnes & Ormandy; Linux UDP-related NULL pointer
  - Pending disclosures in this space
Malware: trends?

- Attackers follow path of least resistance
- Multi-bug payloads
  1. Gain code execution
  2. Escape sandbox
- Two bugs harder than one =>
  - Less 0-day?
  - Increasing black-market exploit value?
- More direct-to-kernel bugs?
  - MS09-065: EOT font parsing
  - MS10-032: TrueType font parsing
  - 3D APIs
Malware: kernel attacks

- Further reading:
  - [Party at ring0](#)

Timeline
Beyond the Sandbox

- Sandboxing is great, but leaves gaps:
  - Sandbox bugs
  - User bugs
  - New APIs poking holes
- Blacklist approaches as defense in depth
  - Mitigate against zero-days
  - Mitigate against phishing, social engineering
Beyond the Sandbox
Beyond the Sandbox

- Key metrics for blacklist approach
  - Freshness of data
  - Coverage
  - Accuracy
Beyond the Sandbox

● Building a blacklist
  o URL discovery
  o Classification
  o Information dissemination
  o Broadening scope (phishing, malware, social engineering)

● Approach varies for phishing, malware
Beyond the Sandbox

- Let's talk about phishing

![Phishing Detected!](image-url)
Beyond the Sandbox

- As we harden the browser + authentication mechanisms, humans remain the weak link
- Phishers obtain compromised credentials, potentially easier than compromising the computer
- Use gmail spam + user submissions to build up list of URLs, machine learning to classify
- go from millions of URLs to a few hundred thousand known patterns at any time
Beyond the Sandbox

- Malware may require a bit more skill, but a zero day can get incredible reach compared to phishing
- Start with billions of URLs (our copy of the web)
- Machine learning to come up with candidate malware sites
- Visit in virtual machine to confirm
Beyond the Sandbox

- Where will the next zero-days lie?
- Many new APIs being added to browser (HTML5++).
- Some APIs expose new devices to the web -- 3d graphics, filesystems, fonts
- May see attacks on drivers now that they are exposed to untrusted web data
- Blacklist based approaches won't save us, but can help mitigate against these new threats