Web Browser Security

Adam Barth

Slides adapted from Collin Jackson

Running Remote Code is Risky

- Integrity
  - Compromise your machine
  - Install malware rootkit
  - Transact on your accounts

- Confidentiality
  - Read your information
  - Steal passwords
  - Read your email

Browser Sandbox

- Goal
  - Run remote web applications safely
  - Limited access to OS, network, and browser data

- Approach
  - Isolate sites in different security contexts
  - Browser manages resources, like an OS
  - Access control: same-origin principle
    - Pages from the "same site" can interact
    - Pages from "different sites" separated

Why study browser security?

... if you’re not Microsoft, Mozilla, Apple, Google or Opera?

- Build better browsers
  - Contribute to open source browsers (Firefox, Safari)
  - Embed a renderer in your program (Gecko, WebKit)

- Build better web applications
  - Servers and firewalls can mitigate browser limitations
  - Take advantage of opt-in browser security features

- Be a safer surfer
  - Make informed security decisions
  - Distinguish harmless warnings from attacks

Threat Models

- Web attacker
  - Controls attacker.com
  - Has HTTPS certificate for attacker.com ($0)
  - User visits attacker.com

- Network attacker
  - Passive: Wireless eavesdropper
  - Active: Evil router, DNS poisoning

- Malware attacker
  - Escaped from browser sandbox

Security User Interface

When is it safe to type my password?
URLs

Global identifiers of network-retrievable documents

http://user:pass@stanford.edu:81/class?name=cs155#homework

Safe to type your password?

Safe to type your password?

Safe to type your password?

Safe to type your password?
Same-Origin Policy
How does the browser isolate different sites?

Are all interactions good?

Browser Same-Origin Policy
- Different origins have limited interaction
  - Origin is the tuple <domain, port, protocol>
    - http://www.example.com:80/whoami  ✓ Full access
    - http://www.example.com:80/hello  ✓ Full access
    - https://www.example.com:443/hello  ✗ Limited access
    - http://www.example.com:443/whoami  ✗ Limited access

Same-Origin Policy Examples
- Example HTML at http://www.site.com/
  `<iframe src="http://othersite.com"></iframe>
  <img src="http://othersite.com/logo.gif">

- Disallowed:
  - `alert(frames[0].document.body.innerHTML)`
  - `alert(frames[0].location)`

- Allowed:
  - `alert(images[0].height)`
  - `frames[0].location = "http://othersite.com/foo";`

Mixed Content
A Guninski Attack

Browser | Policy
---------|---------
IE 6 (default) | Permissive
IE 6 (option) | Child
IE7 (no Flash) | Descendant
IE7 (with Flash) | Permissive
Firefox 2 | Window
Safari 3 | Permissive
Opera 9 | Window
HTML 5 | Child

Legacy Browser Behavior

What should the policy be?

Window Policy Anomaly

Principle: Pixel Delegation

- Frames delegate screen pixels
  - Child cannot draw outside its frame
  - Parent can draw over the child’s pixels
- Navigation similar to drawing
  - Navigation replaces frame contents
  - “Simulate” by drawing over frame
- Policy ought to match pixel delegation
  - Navigate a frame if can draw over the frame
Why Frame Busting Matters

Adoption of Descendant Policy

<table>
<thead>
<tr>
<th>Browser</th>
<th>Policy</th>
</tr>
</thead>
<tbody>
<tr>
<td>IE7 (no Flash)</td>
<td>Descendant</td>
</tr>
<tr>
<td>IE7 (with Flash)</td>
<td>Descendant</td>
</tr>
<tr>
<td>Firefox 3</td>
<td>Descendant</td>
</tr>
<tr>
<td>Safari 3</td>
<td>Descendant</td>
</tr>
<tr>
<td>Opera 9 (many policies)</td>
<td></td>
</tr>
<tr>
<td>HTML 5</td>
<td>Descendant</td>
</tr>
</tbody>
</table>

Intermission

Ask me about:
1000 lines of regression tests
Frame busting and Yahoo
PR for “extended validation” user study

Mashups

How can different sites communicate?

HousingMaps.com

Windows Live Contacts

Integrator

Add a contact
Share contacts
window.postMessage

- Secure channel between frames
  frames[0].postMessage("Attack at dawn!", "http://gadget.com");

- Supported in brand-new browsers

window.addEventListener(function (e) {
  if (e.origin == "http://integrator.com") {
    ... e.data ...
  }, false);

Why include “targetOrigin”?

- What goes wrong?
  frames[0].postMessage("Attack at dawn!");

- Messages sent to frames, not principals
- When would this happen?

Thanks!
You’ve been a great audience