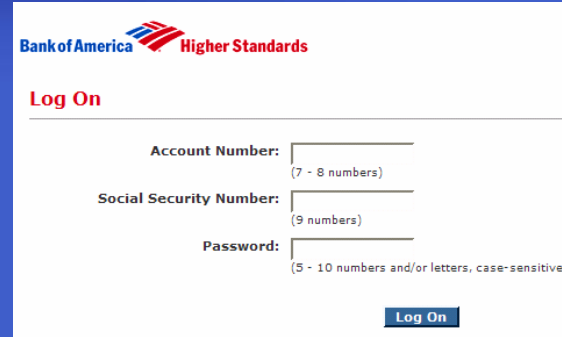



Protecting Browser State from Web Privacy Attacks

Collin Jackson, Andrew Bortz,
Dan Boneh, John Mitchell
Stanford University

Context-aware Phishing

- Bank of America customers see:
- Wells Fargo customers see:
- Works in all major browsers
- Design issue, not a just bug



Bank of America  Higher Standards

Log On

Account Number:
(7 - 8 numbers)

Social Security Number:
(9 numbers)

Password:
(5 - 10 numbers and/or letters, case-sensitive)

Log On



WELLS FARGO 

View Your Accounts

1. Username
[Forgot username?](#)

2. Password
[Forgot password?](#)

3. Sign On to
Account Summary

> Sign On

Example Attacks

- Query visited links:

```
<style>a#visited {  
background: url(track.php?example.com);  
}</style><a href="http://example.com/">Hi</a>
```

- Time browser cache:

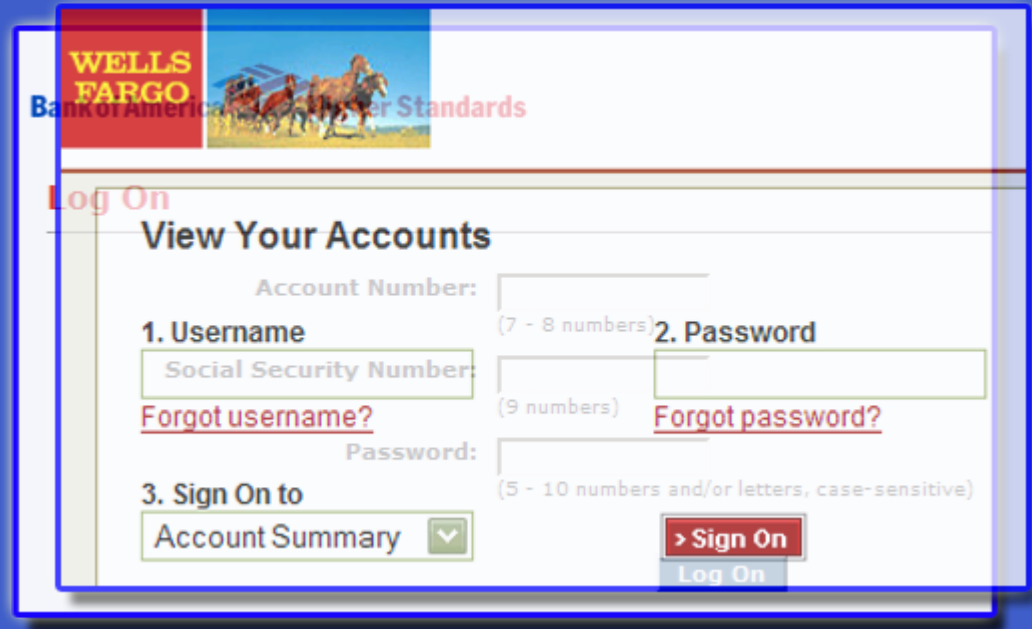
```
<script>start = new Date();</script>  

```

- Can we block script, background image?

Chameleon Pages

- No JavaScript required
- No server involvement
- Even works in Outlook 2002



WELLS FARGO
Bank of America
Higher Standards

Log On

View Your Accounts

Account Number:

1. Username (7 - 8 numbers)

[Forgot username?](#) (9 numbers)

2. Password

[Forgot password?](#)

Password:

3. Sign On to (5 - 10 numbers and/or letters, case-sensitive)

Perspectives

- Phisher: Where do you bank?
- China: Have you been to subversive sites?
- Amazon: Can I show contextual ads?
- Phished site: Can I check history against phishing blacklist?
- PayPal: Can I use history as 2nd factor?
- Sensitive website: Can I protect visitors?
- Browser vendor:



Can I protect users at every site?

Same Origin Principle (strict)

Only the site that stores some information in the browser may later read or modify that information.

- Site: protocol + port + host
- Too restrictive to use in practice
- Web relies on site interconnections

Same Origin Policy (relaxed)

Only the site that stores some information in the browser may later read or modify that information, ***unless it is shared.***

- What is sharing?
- No strict definition
- Relies on expectations of developer/user

Sharing Browser State

- Pass information in query parameters
- Modify `document.domain`
- User permission (IE's trusted zones, Mozilla's UniversalBrowserRead)
- Stylesheets
- Scripts
- Image size
- XMLHttpRequest (not XMLHttpRequest)

```
<script type="text/javascript"><!--  
google_ad_client = "pub-2966125433144242";  
google_ad_width = 110;  
google_ad_height = 32;  
google_ad_format = "110x32_as_rimg";  
google_cpa_choice = "CAAQ_-KZzgEaCHfyBUS9wT0_KOP143Q";  
/--></script>  
<script type="text/javascript" src=  
"http://pagead2.googlesyndication.com/pagead/show_ads.js">  
</script>
```

Inappropriate State Sharing

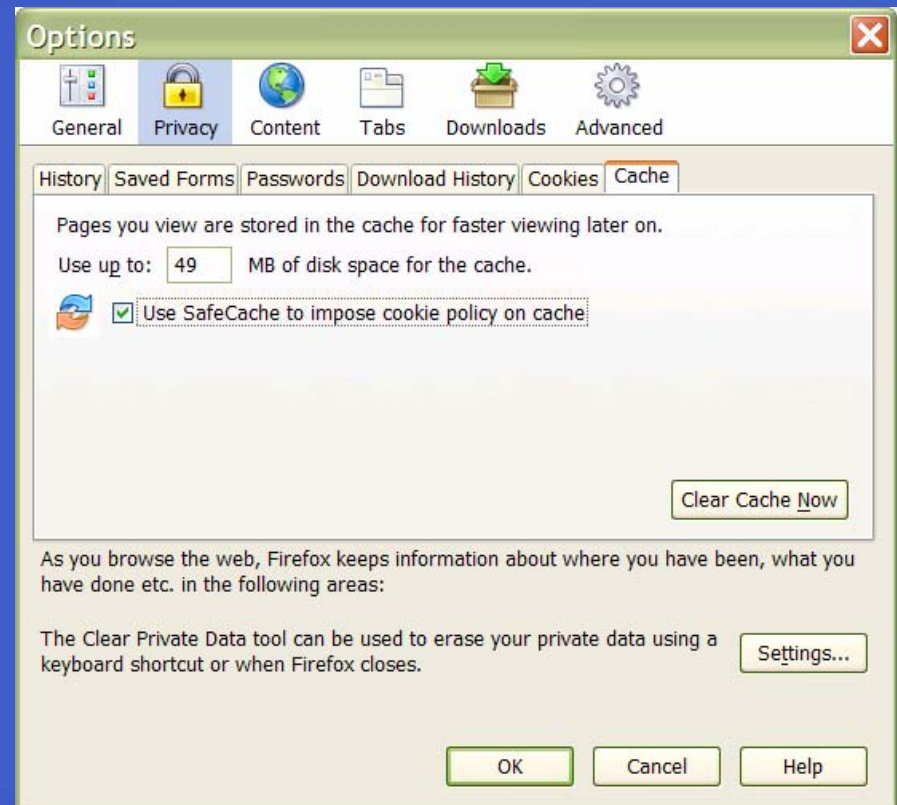
- Common developer/user expectation: browser history is secret



- Options?
 - Change expectations
 - Change browser

SafeCache

- Browser extension for Firefox
- Intercept requests to browser cache
- If no referrer, allow request
- If URL has referrer:
 - Store referrer host with cache entry
 - Cache hit only on referrer host match





SafeHistory

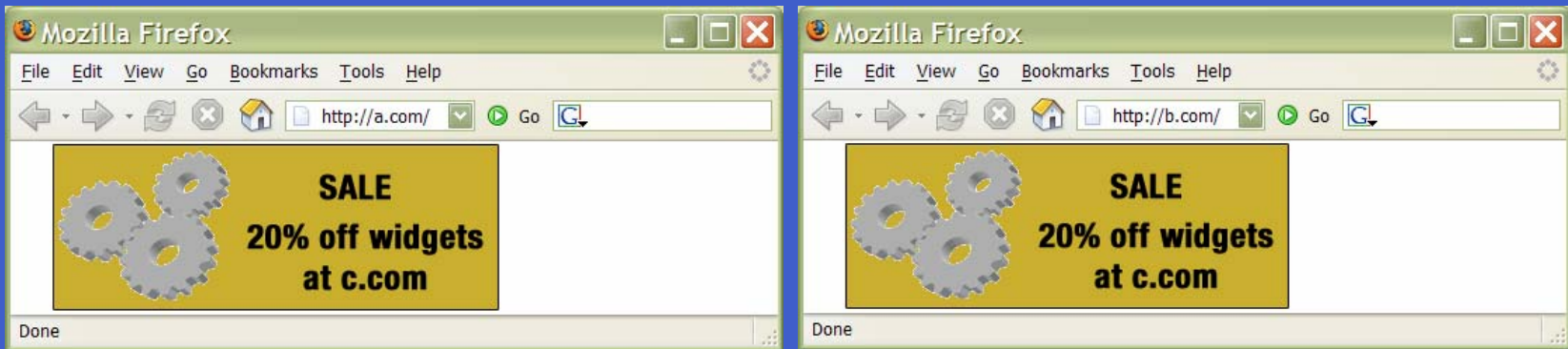
- Intercept requests to browser history database
- For each history entry, record referrers
- Color visited link if:
 - It's a same-site link, or
 - Cross-site link was visited from this site

The screenshot shows the Google search interface. The search bar contains 'stanford university' and the search button is labeled 'Search'. Below the search bar, there are radio buttons for 'the web' (selected) and 'pages from the UK'. The search results section is titled 'Web' and shows 'Results 1 - 10 of about 307,000,000 for [stanford university](#). (0.44 seconds)'. The first result is for 'Stanford University' with a description: 'The Stanford University homepage allows you to find web resources and websites.' Below the description are several links: [www.stanford.edu/](#), [- 26k](#), [- 23 May 2006](#), [- Cached](#), [- Similar pages](#), [Maps](#), [- Admission](#), [- Directions](#), [- Academic Programs](#), and [More results from www.stanford.edu »](#).

The screenshot shows the Yahoo! search interface. The search bar contains 'stanford university' and the search button is labeled 'Search'. Below the search bar, there are several links: [Answers](#), [My Web](#), [Search Services](#), [Advanced Search](#), and [Preferences](#). The search results section is titled 'Search Results' and shows '1 - 10 of about 22,700,000 for [stanford university](#) - 0.13 sec. ([About this page](#))'. Below the search results, there is a section 'Also try:' with links to [stanford university california](#), [stanford university athletics](#), and [More...](#). The first result is for 'Stanford University' with a description: 'Official site includes information about the academic programs, research, admissions, medical center, athletics, student life, and more.' Below the description are several links: [Category: California > Palo Alto > Stanford University](#), [www.stanford.edu](#), and [- More from this site - Save](#).

Third Party Cookies

- Site of embedded image can build history of visitor's activities where image appears.
- IP address is no longer sufficient for tracking



- Solution: Block access to site's own cookies if the domain of the embedding page does not match
- Site accesses own state – not same origin issue

Third Party Blocking Policy





A site may only store or read some persistent information in the browser if it is the same site as the top level page.

- Alternate definition: referrer is same site
- Top level page is the primary interaction
- Storing *or* reading allows tracker to build full record of user's history.

Block on set or read?

- If setting is allowed:
 - Tracker site sets different cookie at every participating site
 - When user visits tracker site in first party context, entire history is visible
- If reading is allowed:
 - Tracker site sets unique user identifier cookie when user visits tracker in first party context
 - When user visits any participating site, tracker updates history database entry on server

Broken Cookie Blocking

					Ideal
Read 3 rd -party cookies	Allow	Block	Allow	Block	Block
Set 3 rd -party cookies	Block	Allow	Block	Allow	Block



Third Party Cache: Example

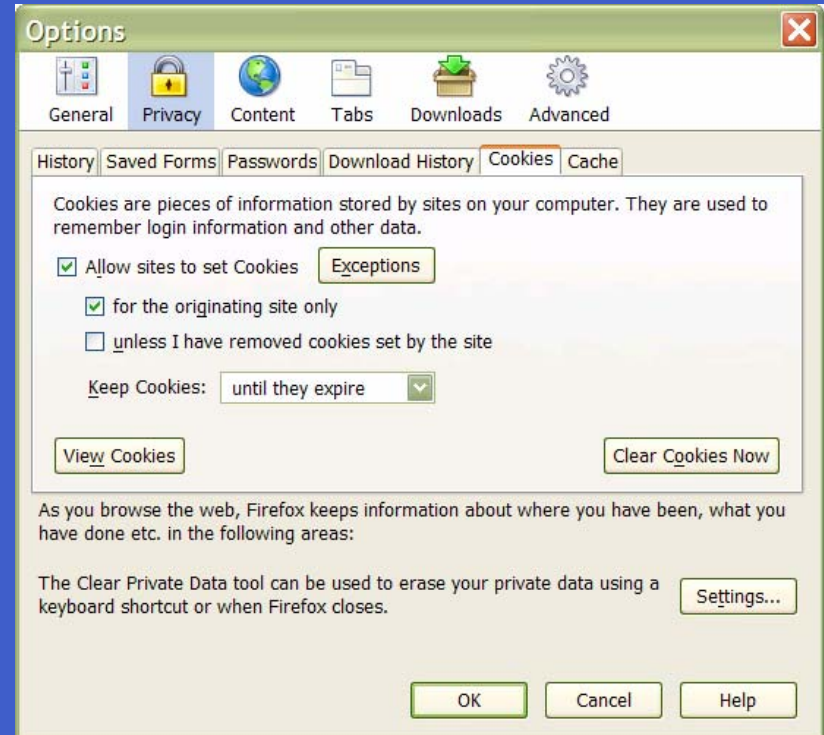
- Offsite script included with `<script src="...">`
- Script generated dynamically and cached

```
<?php /* ----- SERVER-SIDE CACHE DIRECTIVES ----- */
if (getallheaders()['If-Modified-Since']) {           // Check if the browser has a cached copy.
    header('HTTP/1.1 304 Not Modified');             // If so, tell browser to continue using it,
    exit();                                           // and we don't need to send a new identifier.
}                                                     // Otherwise, send cache headers to the browser:
header('Expires: ' . gmdate('D, d M Y H:i:s', time()+365*24*60*60)); // expires one year from today
header('Last-Modified: ' . gmdate('D, d M Y H:i:s', time()));        // content was modified today
$id = rand();                                        // Also, generate a unique identifier for this user.
?> /* ----- CLIENT-SIDE JAVASCRIPT ----- */
var links = document.getElementsByTagName('a');      // Get a list of <a> tags in the current document.
for(var i = 0; i < links.length; i++)              // For each hyperlink found, change the href by
    links.item(i).href += '?userid=<?php echo $id ?>'; // appending the server-generated user id to the end.
```

- Unique identifier now appended to all links

General Third Party Blocking

-  SafeCache:
Disallow cache for
offsite content
-  SafeHistory:
Show links as
unvisited in cross-site
frames



Bypassing Third Party Blocking

- Protects sites from each other
- Many covert channels if sites cooperate
 - JavaScript redirection
 - Meta refresh
 - Popup windows
 - Cross-site hyperlinks
- Certain techniques are implicit cooperation
 - Frames, scripts, CSS can have active content
- Defense: Disable or clear persistent state

Summary

- Same origin policy: critical for security
 - Restricts cross-site state access
- Third party blocking: additional privacy
 - Restricts site's access to its own state
 - Incorrectly implemented in all major browsers
 - Most effective for images
- Neither technique stops cooperative sharing



safecache.com



safehistory.com

