



A Laugh **RIAt**

Security in Rich Internet Applications



Rafal M. Los

HP ASC Sr. Security Solutions Expert

HP SOFTWARE

CAUTION

**THIS SIGN HAS
SHARP EDGES**

DO NOT TOUCH THE EDGES OF THIS SIGN



ALSO, THE BRIDGE IS OUT AHEAD



Now Hear This



Hacking is
illegal

You should
only try
this at home
on your own
code

I encourage
you to think

Now Hear This

BUT...

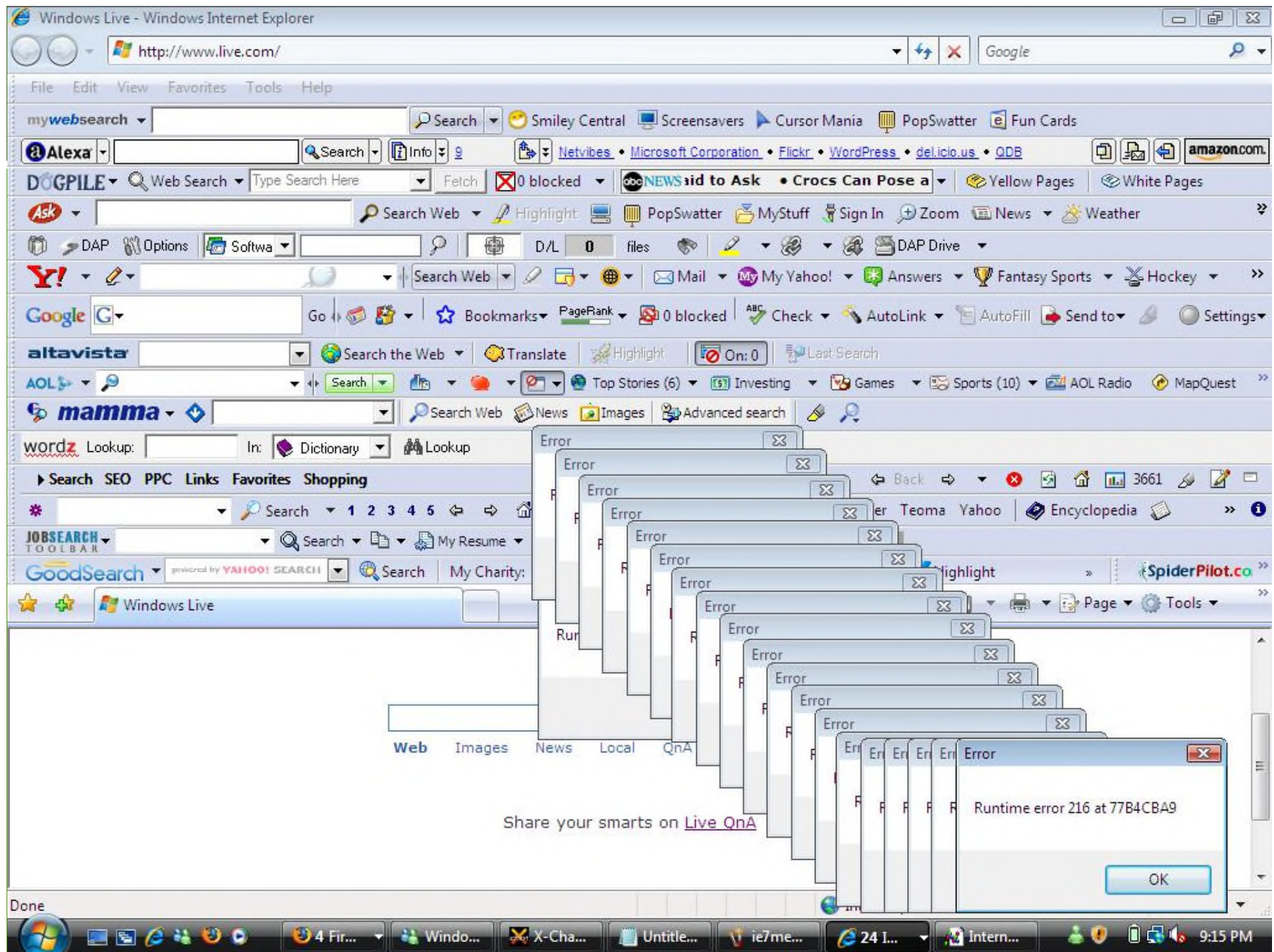
- Rich Internet Apps are taking over
- Pendulum is too far to *functionality*
- Security is more important now than ever
- Developers are writing terrible code
- ... and the bad guys are making money off your flaws



Define: R.I.A.

Ask Wikipedia...

Rich Internet applications (RIAs) are **web applications that have some of the characteristics of desktop applications**, typically delivered by way of a proprietary web browser plug-ins or independently via sandboxes or virtual machines



Browser Overcrowding

- The browser's main purpose was to render HTML... and scripting languages...
- RIA via plug-ins...
 - *enhance* user experience
 - push additional functionality to the browser
 - ...migrate server function to desktop
- **RIA is a straight-on disaster!**

Why the Browser?

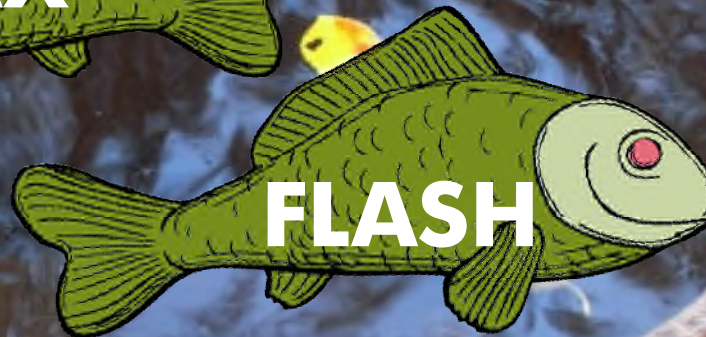
Moving server-side *functionality* to the client is causing... challenges for security

- Exposed APIs (a la AJAX)
- Client-side logic
- Visual technologies add scripting
- Client is defenseless
- Client can be 100% manipulated

Does this mean RIA is 100% bad?

- **YOU** decide...

Fish in a Barrel



simple to analyze
informative
transparent'ish

First a Word on RIA

- Rich Internet Applications **do not**
 - Produce a new class of vulnerabilities
 - Make it impossible to *secure* the code
- Rich Internet Applications **do**
 - Create a massive new attack surface
 - Potentially make small coding mistakes epic
 - Move server-side (hidden) function to the client (exposed)

Target: AJAX

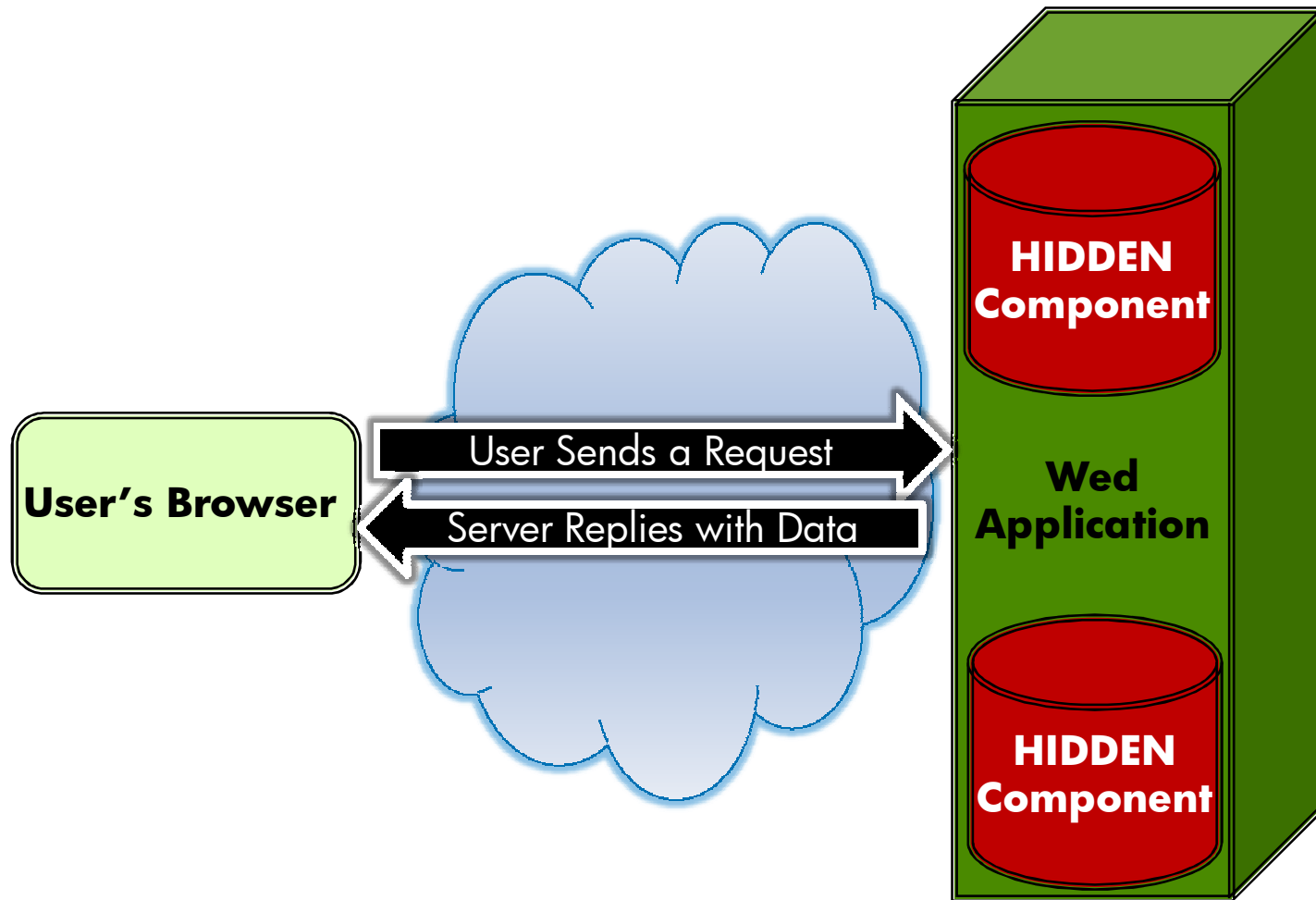
Wikipedia definition:

AJAX (Asynchronous JavaScript and XML), is a group of interrelated web development techniques... With Ajax, **web applications can retrieve data from the server asynchronously in the background** *without interfering* with the display and behavior of the existing page

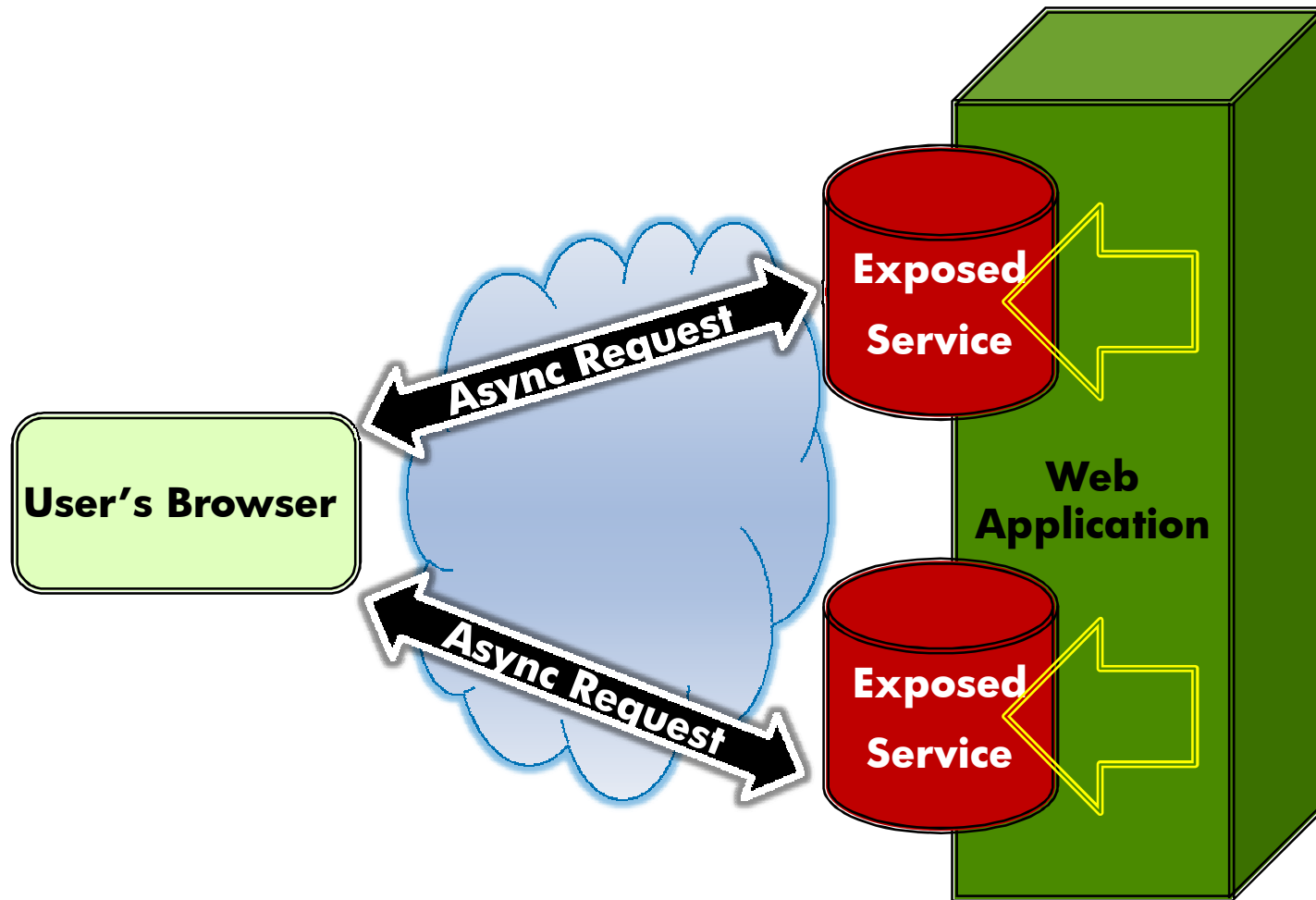
AJAX at 50,000ft

- Application Logic moved out to the client
- Allows for a rich user experience
 - No full-frame browser refreshes
 - Only pieces of the “page” have to refresh
 - Asynchronous fetch
 - No need to send... wait... render anymore!
 - User-independence
 - Data fetched *as needed* by the framework
 - Goes way beyond boring HTML
 - Highly interactive applications

Target: AJAX



Target: AJAX



AJAX is really neat...

**LET'S USE THIS TO OUR
ADVANTAGE!**

Target: AJAX

XMLHttpRequest Object

- Part of the DOM API
- Implemented *differently* in each browser
- Interact directly with web server
- No need for user interaction
- Modify the active document without reloading the entire page

Target: AJAX

Example: MapQuest.com

- Scrolling through the map...
- Browser makes requests for you
 - <http://www.mapquest.com/dwr/call/plaincall/HomeFormService.getWeatherSummary.dwr>
 - <http://www.mapquest.com/dwr/call/plaincall/AdServiceProxy.makeAdCall.dwr>
- Browser auto-fetches requests without your input

Target: AJAX

Let's dissect what's going on...

- <http://www.mapquest.com/dwr/call/plaincall/HomeFormService.getWeatherSummary.dwr>
- <http://www.mapquest.com/dwr/call/plaincall/AdServiceProxy.makeAdCall.dwr>
- <http://www.mapquest.com/dwr/call/plaincall/AuthService.autoLogin.dwr>
- At least 3 exposed services
 - HomeFormService
 - AdServiceProxy
 - AuthService
- Exposed functions
 - HomeFormService → getWeatherSummary
 - AdServiceProxy → makeAdCall
 - AuthService → autoLogin

Target: AJAX

<http://www.mapquest.com/dwr/call/plaincall/HomeFormService.getWeatherSummary.dwr>

POST data

```
callCount=1 page=/ httpSessionId= scriptSessionId=sessionId639 c0-
scriptName=HomeFormService c0-methodName=getWeatherSummary c0-id=0 c0-
e1=number:42.103298 c0-e2=number:-88.372803 c0-e3=null:null c0-e4=null:null c0-
e5=string:Gilberts c0-e6=string:IL c0-e7=null:null c0-e8=string:US c0-e9=string:CITY
c0-param0=Object_Object:{latitude:reference:c0-e1, longitude:reference:c0-e2,
id:reference:c0-e3, addressLine1 :reference:c0-e4, city:reference:c0-e5,
state:reference:c0-e6, postalCode:reference:c0-e7, country:reference :c0-e8,
geocodeQuality:reference:c0-e9} batchId=0
```

Response set

```
//#DWR-INSERT // #DWR-REPLY var
s0={};s0.dewPoint=null;s0.dewPointUnits=null;s0.forecasts=null;s0.humidity=null;s0.hu
midityUnits
=null;s0.icon="http://deskwx.weatherbug.com/images/Forecast/icons/cond002.gif";s0.nam
e=null;s0.shortTitle ="Partly
Cloudy";s0.station=null;s0.temperature=47.0;s0.temperatureUnits="F";s0.windDirection=
null;s0 .windSpeed=null;s0.windSpeedUnits=null;s0.zip="60102";
dwr.engine._remoteHandleCallback('0','0',{data:s0,detailCode:null,errors:null,statusC
ode:"SUCCESS"}) ;
```

Target: AJAX

- How would you approach the previous example?
 - Enumerate as many services as possible
 - Identify as many methods as possible
 - Push various data sets to gleam results
- Let's do a practical example!
 - FireFox
 - Firebug
 - Favorite intercepting proxy
 - RAW http editor

Example: MapQuest

- Let's search for cheap gas!

Our proxy captures this interesting request...

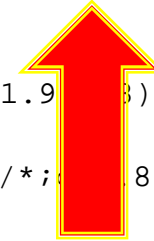
```
http://gasprices.mapquest.com:80/searchresults.jsp?search=true&latitude=&longitude=&gasPriceType=3%2C4%2C5&address=5260+morningview+drive&city=hoffman+estates&stateProvince=IL&postalCode=99999&radius=0&brand=&sortOrder=2
```

- Let's analyze that a little further...
can we manipulate it somehow?

Example: MapQuest

RAW Request

- GET
/searchresults.jsp?search=true&latitude=&longitude=&gasPriceType=3%2C4%2C5&address=52
60+morningview+drive&city=hoffman+estates&stateProvince=IL [REDACTED] radius=0&
brand=&sortOrder=2 HTTP/1.1
- Host: gasprices.mapquest.com
- User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.9.1.3) Gecko/2009032609 Firefox/3.0.8 (.NET CLR 3.5.30729)
- Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
- Accept-Language: en-us,en;q=0.5
- Accept-Encoding: gzip,deflate
- Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
- Keep-Alive: 300
- Referer: http://gasprices.mapquest.com/
- Cookie: locationhistory="{42109700 -88366667 ADDRESS {182 Charleston Ln} Gilberts IL 60136-8027 {} US {} {} 6} {41886820 -87627118 ADDRESS {35 E Wacker Dr} Chicago IL 60601-2314 {} US {} {} 6} {42103298 -88372803 CITY {} Gilberts IL {} {} US {} {} 6} {41682800 -88351402 CITY {} Oswego IL {} {} US {} {} 6}"; s_cc=true; s_sq=aolwpmq%2Caolsvc%3D%2526pid%253Dgasprices%252520%25253A%252520gasprices%252520%25253A%252520gaspricesweb.home%2526pidt%253D1%2526oid%253Dfunctiononclick%252528event%252529%25257B%252520%252520document.getElementById%252528%252522search%252522%252529.value%25253Dtrue%25253B%25257D%2526oidt%253D2%2526ot%253DIMAGE; tsession=PlhFWXOcnlKzUH/X8nB+O8ZJlbY=



Example: MapQuest

- Simple script injection!

- **Original:**

`http://gasprices.mapquest.com/searchresults.jsp?search=true&latitude=&longitude=&gasPriceType=3,4,5&address=5260+morningview+drive&city=hoffman+estates&stateProvince=IL&postalCode=60192&radius=0&brand=&sortOrder=2`

- **Manipulated:**

`http://gasprices.mapquest.com/searchresults.jsp?search=true&latitude=&longitude=&gasPriceType=3,4,5&address=5260+morningview+drive&city=hoffman+estates&stateProvince=IL&postalCode=""><frame src=http://google.com></iframe><script>alert(document.cookie)</script>&radius=0&brand=&sortOrder=2#93936520642628051000`

Example: MapQuest

The screenshot shows a web browser window displaying the MapQuest website. The address bar contains the URL: `http://gasprices.mapquest.com/searchresults.jsp?search=true&latitude=&longitude=&gasPriceType=3,4,5&address=5260+morningview+drive&city=hoffm`. A red box highlights a JavaScript code snippet in the browser's address bar, which is a snippet of a larger script. The code snippet is:

```
co=usa;"; //Dma adSetOthDclk(dmaStr); //Magic Number var magicnumber=search; var magicnumber_top=search_top; var magicnumber_right=search_right; //MapSettings
var iSortOrder = 2; var iGasPriceTypeSort = 3; var sGasPriceType = "3,4,5"; var aGasPriceType = new Array(3,4,5); var mqTileMap, mqViewControl, mqZoomControl; var
mqPoiCollection = new MQPoiCollection(); var mqPoi, mqMapIcon; var mqOriginPoi = null; mqOriginPoi = new MQPoi(new MQLatLng(42.062197, -88.209724));
```

Below the address bar, the MapQuest logo is visible, along with navigation links for Maps, Directions, Yellow Pages, Local, and Gas Prices. A banner for "Prices for HOFFMAN ESTATES, IL" shows a "Lowest" price of \$1.89 and a "Highest" price of \$2.13, with a "Gas Calculator" link. A "Find Gas Prices" form is present, with fields for "Gasoline", "Address or Intersection" (5260 morningview drive), "City" (hoffman estate), "State" (IL), and "ZIP Code" (60192). A red arrow points to the `tabindex="5"` attribute in the "City" field. An "Alert" dialog box is open, displaying the URL `http://gasprices.mapquest.com/` and the text `s_cc=true; s_sq=%5B%5BB%5D%5D`. A red box highlights the text in the alert dialog. The dialog also includes a checkbox for "Prevent this page from creating additional dialogs." and an "OK" button.

Thoughts: AJAX

The screenshot shows the Firebug interface with the 'Net' tab selected. A red box highlights the 'XHR' tab in the top navigation bar. Another red box highlights a request to `http://www.airplay.com/getschedule.svc?size=5&sorted=desc&startday=20090327`. A third red box highlights the 'Response' tab, which shows the following headers:

Response Headers

- Date: Fri, 03 Apr 2009 16:19:28 GMT
- Server: Apache/2.2.3 (CentOS)
- Vary: Accept-Encoding
- Content-Encoding: gzip
- Connection: close
- Transfer-Encoding: chunked
- Content-Type: text/xml
- Via: 1.1 SPI
- Proxy-Connection: Keep-Alive


Request Headers

- Host: www.airplay.com
- User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.9.0.8) Gecko/2009032609 Firefox/3.0.8 (.NET CLR 3.5.30729)
- Accept: text/javascript, text/html, application/xml, text/xml, */*
- Accept-Language: en-us,en
- Accept-Encoding: gzip, deflate
- Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.3
- Keep-Alive: 300
- Proxy-Connection: keep-alive
- X-Requested-With: XMLHttpRequest
- X-Prototype-Version: 1.6.0
- Referer: http://www.airplay.com
- Cookie: Coyote=2

Annotations on the screenshot:

- A box with the text "Use FireBug - look for NET -> XHR requests" points to the 'Net' tab.
- A box with the text "Enumerate exposed services and APIs" points to the request URL.

A large green box at the bottom contains the text: "AJAX is everywhere... learn to spot it".



Let's shift gears

ADOBE FLASH! [SWF]



EPIC FAIL

Seriously, how the did you manage that?

Target: Flash

- What do we know about a flash object?
 - Compiled objects (not human-readable)
 - Bi-directional multimedia streaming and presentation (audio/video)
 - Scriptable functionality via ActionScript
 - Being used to build highly interactive applications
 - ~~Secure~~

Target: Flash

- Hack Flash? Why?
 - FREE stuff
 - "Billy wins a cheezeborger"
 - http://www.youtube.com/watch?v=_bHtGD3qUVg
 - Steal data(bases)
 - Database access from flash!
 - <http://code.google.com/p/assql/>
 - *"asSQL is an Actionscript 3 Mysql Driver aimed towards AIR projects to allow Mysql database connectivity directly from Actionscript"*
 - Steal confidential information
 - Hidden passwords? Secret URLs... etc!

Target: Flash

- Flash is semi-transparent
 - You can decompile it! (mostly)
 - Many good de-compilers exist
 - SWFScan (HP's free tool):
<https://h30406.www3.hp.com/campaigns/2009/wwcampaign/1-5TUVE/index.php?key=swf>
 - Flash de-constructor resources:
<http://tinyurl.com/cgbkqn>
- Source code reveals secrets
 - People hide passwords
 - Database connection strings
 - Encryption keys
 - ...etc!

Target: Flash

- Google'ing for good flash to *examine*
 - Query: `inurl:login filetype:swf`
 - Query: `inurl:play filetype:swf`

[FLASH] [CLICK TO PLAY! 0 score time left 0 0 score time left 30 0 score ...](#)  

File Format: Shockwave Flash


3. 0. score. time left. 2. 0. score. time left. 1. PLAY AGAIN? SEND. 0. YOU SCORED. First Name: Congratulations High Score! Your Email: ...



[play.sockandawe.com/flash/shootinggallery.swf](#) - [Similar pages](#) - 

[FLASH] [I lo loa load loadi loadin loading Production play it again Kids ...](#)  

File Format: Shockwave Flash

I lo loa load loadi loadin loading Production play it again Kids! Hang in there, Almost there... Toothy Knowin' Yal Spin Fun End The forget to Floss! ...

[spoznajigre.play-euro-lotto.com/1-Spin%20Fun%20Knowing%20Ya.swf](#) - [Similar pages](#) - 

[FLASH] [05% ROUND OVER TOTAL - BONUS - SCORE - SCORE - TOTAL - This is a ...](#)  

File Format: Shockwave Flash

05% ROUND OVER TOTAL - BONUS - SCORE - SCORE - TOTAL - This is a big dea Hit 69 Plates to get to the next level. Level 7: Winter Wonderland HIT SPACE BAR TO ...

[www.cbsgames.com/games/play/skeet-shoot_cbs1.swf](#) - [Similar pages](#) - 

[FLASH] [LOADING -- FREEONLINEGAMES.COM GAMES.COM ...](#)  

File Format: Shockwave Flash

LOADING --. FREEONLINEGAMES.COM. GAMES.COM. ONLINE. FREE. 0. You ... Restart ... Play More Games. - Unbeaten. Zoran Primorak ...

[www.thatwasrandom.com/play/swf/tabletennis.swf](#) - [Similar pages](#) - 



Target: Flash

Sometimes... you get this

<SNIP>

```
on (release, keyPress '<Enter>') {  
    if (password eq 'Devlin778') {  
  
        getUrl('http://www.SomeCompany.tld/client_pages/CUSTOMER_REMOVED/778.html  
' , '');  
    } else {  
        if (password eq 'Maginness781') {  
  
            getUrl('http://www.SomeCompany.tld/client_pages/CUSTOMER_REMOVED/781.html  
' , '');  
        } else {  
            if (password eq '783-1') {  
  
                getUrl('http://www.SomeCompany.tld/client_pages/CUSTOMER_REMOVED/783.html  
' , '');  
            } else
```

</SNIP>

Target: Flash

And if you're lucky...

```
private static function query(arg0:String, arg1:flash.events::EventDispatcher = null)
{
    st = null;
    token = null;
    statement = arg0;
    dispatcher = arg1;
    trace("2:MySQL Query: " + statement);
    if(this.connection == null)
    {
        try {
            this.connection = new Connection(irrcrpt("dggurjudgh.frp", 3), 3306, irrcrpt("icog_nqikp",
2), irrcrpt("dlsu4y", 1), irrcrpt("jdph", 3));

        } catch (e:SecurityError) {
            var loc1:* = e;
            statement = null;
            Alert.show(statement.message, "Security Error");
            if(dispatcher)
            {
                dispatchEvent(new Event(Event.CANCEL));
            }
            return;
        }
    }
}
```



Target: Flash

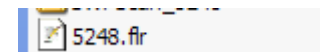
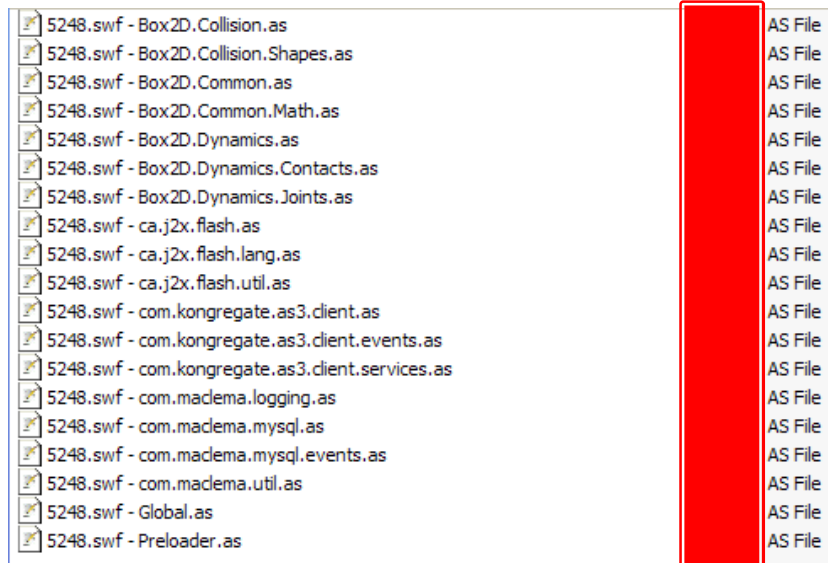
pwning in a Flash

- Discovery
 - search, identify target
- Decompile/deconstruct
 - Critical to get a good decompiler
 - There is a HUGE difference
 - Pull out all ActionScript
- Analyze
- Repurpose | reconstruct
- Exploit...



Target: Flash

- Not all de-compilers are alike
 - SWFScan is thorough!
 - 19 object source files
 - 1.02Mb total code
 - Flare isn't...
 - 1 object source file
 - 2kb total code



Target: Flash

- You've got source, now what?
 - Look for *interesting* things
 - Database connection strings
 - Connection constructors (sending data)
 - Password validation
 - "Hidden" data (coupon codes, options)
 - Re-purpose the code
 - Create an application as a front-end to DB
 - Create a "push button and win" game
 - Other less evil alternatives...

Target: AdultSwim

- Let's check out a game
 - "ZombieHookerNightmare" from AdultSwim.com
- Purpose:
 - Get the high score, get on TV (fame)
- Approach:
 - Download, deconstruct, FTW



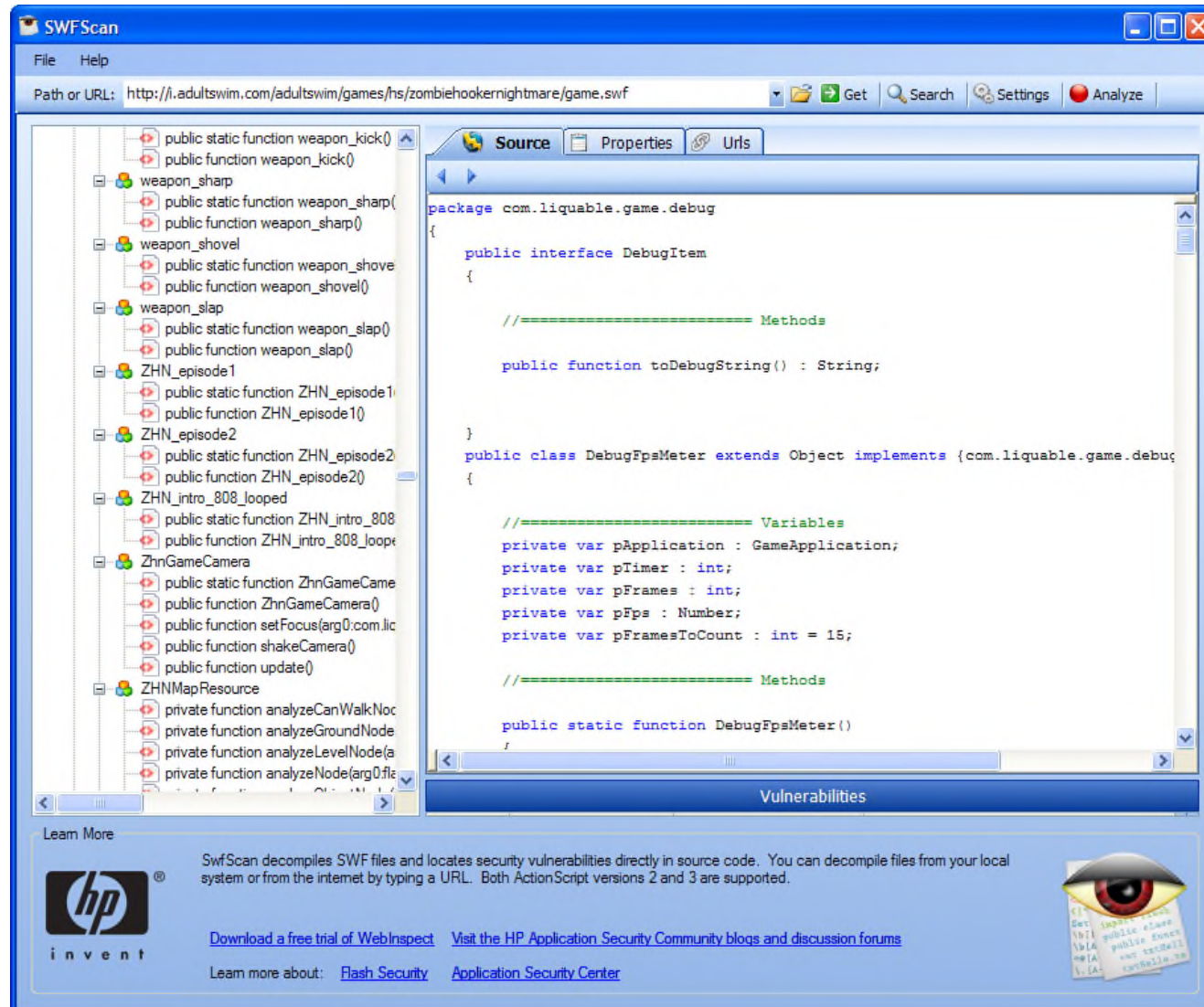
Target: Adult Swim

- Acquire Target...

```
GET /adultswim/games/hs/zombiehookernightmare/game.swf HTTP/1.1
Host: i.adultswim.com
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.9.0.8)
    Gecko/2009032609 Firefox/3.0.8 (.NET CLR 3.5.30729)
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip,deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
Referer:
    http://www.adultswim.com/games/game/index.html?game=zombiehookernightmare
Cookie: s_cc=true; s_sq=%5B%5B%5D%5D; s_vi=[CS]v1|49D6898000004868-
    A3A083600000AB6[CE];
    adDEmas=R00&hi&sbcglobal.net&73&usa&602&60601&14&07&U1&M1&105&;
    adDEon=true
```





























Target: Adult Swim

- Disassemble



Target: Adult Swim

- Wade through *tons of code*

 game.swf - breakable_objects fla.as	2 KB	AS File
 game.swf - com.liquable.data.as	1 KB	AS File
 game.swf - com.liquable.game.as	17 KB	AS File
 game.swf - com.liquable.game.astar.as	8 KB	AS File
 game.swf - com.liquable.game.camera.as	7 KB	AS File
 game.swf - com.liquable.game.debug.as	11 KB	AS File
 game.swf - com.liquable.game.events.as	3 KB	AS File
 game.swf - com.liquable.game.geom.as	7 KB	AS File
 game.swf - com.liquable.game.postoffice.as	14 KB	AS File
 game.swf - com.liquable.game.tiledmap.as	41 KB	AS File
 game.swf - com.liquable.game.tiledmap.events.as	3 KB	AS File
 game.swf - com.liquable.game.tiledmap.resources.as	27 KB	AS File
 game.swf - com.liquable.game.tiledmap.resources.events.as	3 KB	AS File
 game.swf - com.liquable.game.ui.as	12 KB	AS File
 game.swf - com.liquable.media.as	8 KB	AS File
 game.swf - com.liquable.statemachine.as	5 KB	AS File
 game.swf - com.liquable.utils.as	16 KB	AS File
 game.swf - com.liquable.visualstack.as	5 KB	AS File
 game.swf - combos fla.as	1 KB	AS File
 game.swf - Global.as	368 KB	AS File
 game.swf - HUD01 fla.as	6 KB	AS File
 game.swf - lib_decomposedZombie fla.as	4 KB	AS File
 game.swf - lib_genericZombie fla.as	3 KB	AS File
 game.swf - lib_lola fla.as	18 KB	AS File
 game.swf - lib_purple_decomposedZombie fla.as	6 KB	AS File
 game.swf - lib_thrownWeapon fla.as	1 KB	AS File
 game.swf - lib_trailer fla.as	3 KB	AS File
 game.swf - screens fla.as	6 KB	AS File

- 28 total files

- 591 Kb of source

Target: Adult Swim

- Yahtzee

```
public static function submit(arg0:String, arg1:Number) : String
{
    strURI = ExternalInterface.call("getLittleServer");
        nGameId = gameId;
        nScore = score;
        nTime = ExternalInterface.call("getSrvrTime");
        strTime = toString();
        strN1 = substr(253, 3);
        strN2 = substr(252, 3);
        n1 = parseInt(strN1);
        n2 = parseInt(strN2);
        nAlgo = n1 * n2 * nScore + nScore;
        strToPass = nGameId + "," + nScore + "," + nTime + "," + nAlgo;
        encrypted_data = MD5.hash(strToPass);
        submission_data = "score=" + nScore + "|gameId=" + nGameId + "|timestamp=" + nTime +
        "|key=" + encrypted_data;
        variables = new URLVariables();
        variables.attr1 = submission_data;
        request = new URLRequest(strURI);
        request.data = variables;
        navigateToURL(request, "_self");
        return submission_data;
}
```

Target: Adult Swim

- What does this function tell us
 - Everything we need to know to get the “high score” posted to the server
- “Faking” a high score
 - Pick a high score you want
 - Run the function
 - Submit a fake score
 - FTW?

Target: Adult Swim

FTW

1. Focus :

```
submission_data = "score=" + nScore + "|gameId=" +  
nGameId + "|timestamp=" + nTime + "|key=" +  
encrypted_data
```

2. Generate *encrypted data*

```
n1 = parseInt(strN1);  
n2 = parseInt(strN2);  
nAlgo = n1 * n2 * nScore + nScore  
encrypted_data = MD5.hash(strToPass);  
strToPass = nGameId + "," + nScore + "," + nTime + "," +  
nAlgo;
```

3. Send string to server!

```
GET /highscores/SubmitScoreServlet.do?attr1=score...
```

Target: Adult Swim

What it looks like on the wire

GET

/highscores/SubmitScoreServlet.do?attr1=score%3D5090%7CgameId%3D1855%7Ctimestamp%3D1238800280000%7Ckey%3D352f27285674930a0257bde0bae32f82

HTTP/1.1

Host: highscores.adultswim.com

User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.9.0.8) Gecko/2009032609 Firefox/3.0.8 (.NET CLR 3.5.30729)

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8

Accept-Language: en-us,en;q=0.5

Accept-Encoding: gzip,deflate

Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7

Keep-Alive: 300

Referer: http://www.adultswim.com/games/game/index.html?game=zombiehookernightmare

Cookie: <cookie stuff>

Target: Flash

- Lessons learned?
 - Don't store sensitive information in Flash objects
 - When deconstructing, get a good decompiler
 - Remember... *encryption* only works when the key is actually secret
 - Know what to look for when auditing
- For flash... stick to games/video



Wrapping Up

Rich Internet Applications [RIA] are dangerous if misunderstood

- RIA: bottom line
 - NO additional vulnerability types
 - MASSIVE additional attack surface

The client is **never** a safe place

Don't learn to hack, hack to learn

Seriously, Though

It's all about **RISK**...

Can you quantify RIA $\rightarrow f(\text{risk})$?

What are the components of risk?

Look beyond *vulnerabilities*

Change your point of view

Learn a different language

Bottom Line: If you talk, does management understand you?

Special thanks to everyone who submitted ideas and voted on "*Name That Talk*"... and the winner is-

ZACH LANIER – AKA “QUINE”

ZACH RUNS “SECURITY TWITS” ON TWITTER...
FOLLOW @QUINE TO GET IN ON GREAT INFOSEC NEWS...

Special Thanks



Rob Fuller aka "Mubix"
Steve Ragan
Mike Bailey
Zach Lanier aka "Quine"
Jeff Brinskelle
Rob Ragan
Billy Hoffman



Rafal Los – “Raf”

HP/ASC – Security Evangelist & Solution Architect

Twitter: <http://twitter.com/RafalLos>

Main Blog: <http://preachsecurity.blogspot.com>

HP Blog: <http://www.communities.hp.com/securitysoftware/blogs/rafal>