Greater Than One

Defeating "strong" authentication in web applications

- Brendan O'Connor

Introduction

- Background Information
- Control Types
- Device Fingerprinting
- One Time Passwords
- Knowledge Base Archives
- Conclusions

Introduction

- Internet Banking
  - Bill Pay
- Car Loans and Mortgages
- Retirement Plans / 401K
- Stock Trading / Investments

Background

Federal Financial Institution Examination Council

Authentication in an Internet Banking Environment

The agencies consider single-factor authentication, as the only control mechanism, to be inadequate for high-risk transactions involving access to customer information or the movement of funds to other parties . . . Account fraud and identity theft are frequently the result of single-factor (e.g., ID/password) authentication exploitation.

source: http://www.ffiec.gov/ffiecinfobase/resources/info_sec/2006/frb-sr-05-19.pdf

Background

- Access to customer information or movement of funds – read: pretty much every screen in an Internet Banking application
- Does not mandate 2 factor authentication – says that single factor is insufficient (greater than one)
- Hardware tokens are expensive and easily lost or broken
- Biometrics for the end user are out of the question

Control Types

- Mutual Authentication
- Device Fingerprinting
- Out of Band Authentication
- One Time Passwords
- Knowledge Base Archives

Control Types

- Mutual Auth
  - This is not device based Mutual Auth
  - Site to user authentication
- Device Fingerprinting
  - Persistent cookies
  - Information from HTTP headers
  - Device Interrogation

Control Types

- Out of Band Auth
  - Not true OOB Auth
  - Only delivery is Out of Band. Authentication still happens within HTTP session
  - Email delivery, SMS message to cell phone, Phone call that reads you a PIN

Control Types

- One Time Passwords
  - Dynamic single use password or PIN (generally delivered via OOB method)
  - Static pre-issued One Time Pads
  - Not to be confused with algorithmic token based auth (such as RSA SecurID©)

Control Types

- Knowledge Base Archives (KBAs)
  - Questions based on information gleaned from public records databases
  - In 2002 did you buy:
    1. Honda Accord
    2. Toyota Camry
    3. Ford Taurus
    4. None of the Above

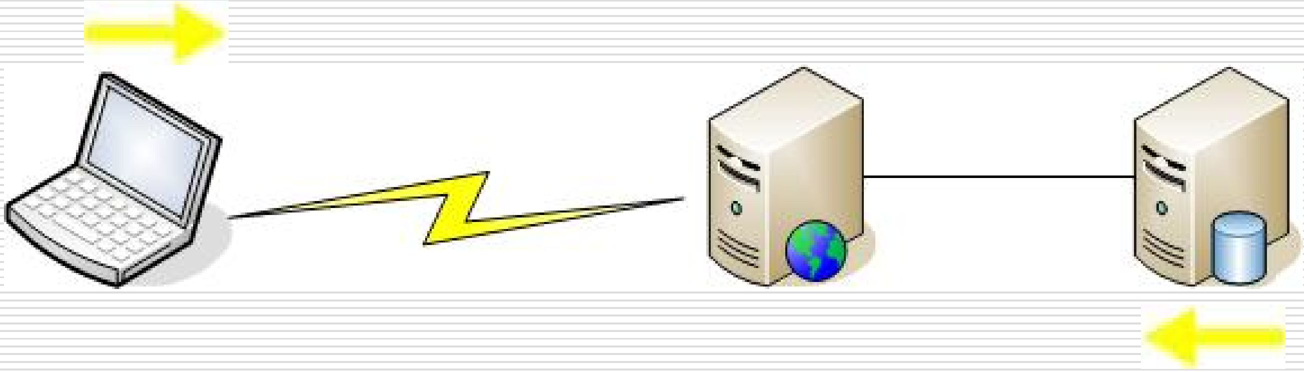
Control Types

- Bolt On vs. Built In
- Enhanced authentication is usually a third party product integrated into existing application
  - Increased attack surface
  - Standard authentication process must be interrupted
  - Exploit architectural weaknesses

Authentication Architecture

Simple Request/Response Authentication

1. Post username/password
2. Database lookup
3. Return 1 or 0
4. “Invalid username or password”

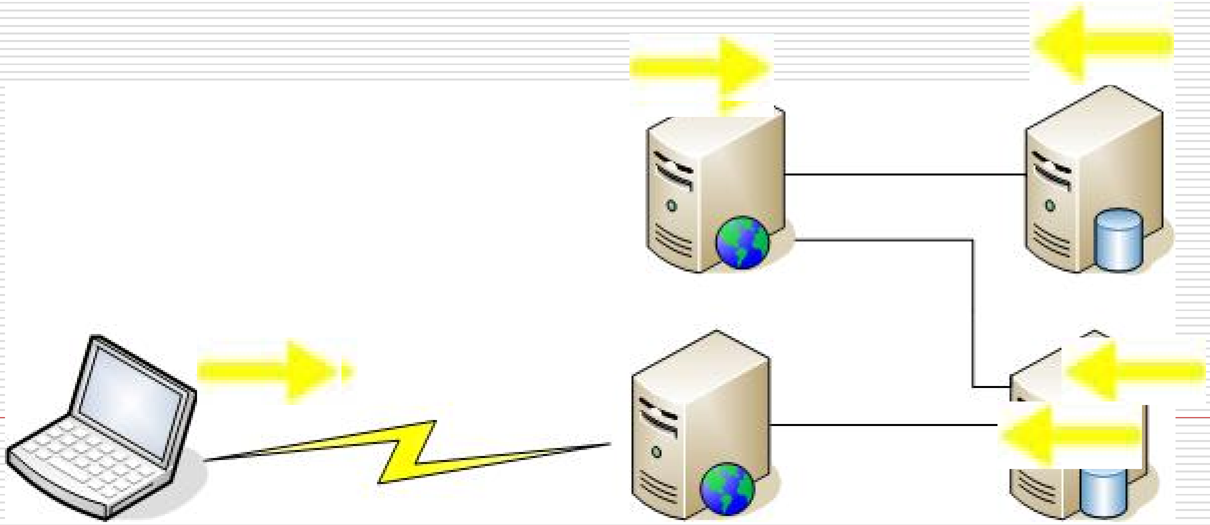


Device Fingerprinting

- Hybrid Approach
  - Picture/phrase based mutual auth
  - OTP or challenge questions required if device is not recognized
  - Persistent cookie set after passing auth criteria
- Request Analysis
  - Single server or multiple server authentication

Device Fingerprinting Request Flow

1. Push auth to new system
2. Valid user?
3. Match auth criteria? (cookie, fprint)
4. Challenge questions/OTP
5. Success – Resume authentication
6. Logged In

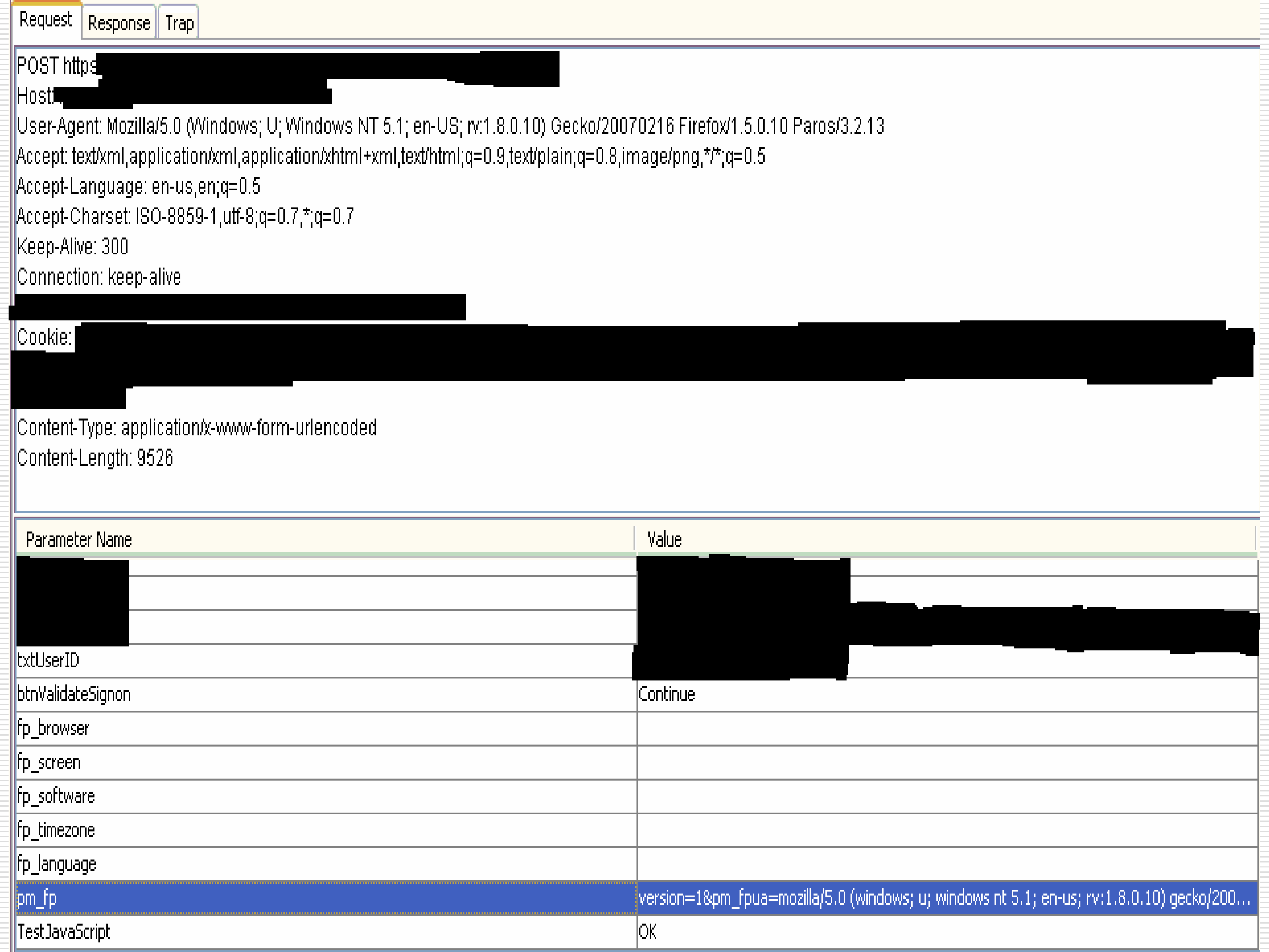


Authentication Flow

- Post username (and cookie if exists)
- Challenge for device fingerprint
- Post Fingerprint (if no cookie)
- New Authentication challenge
- Answer challenge
- Old login

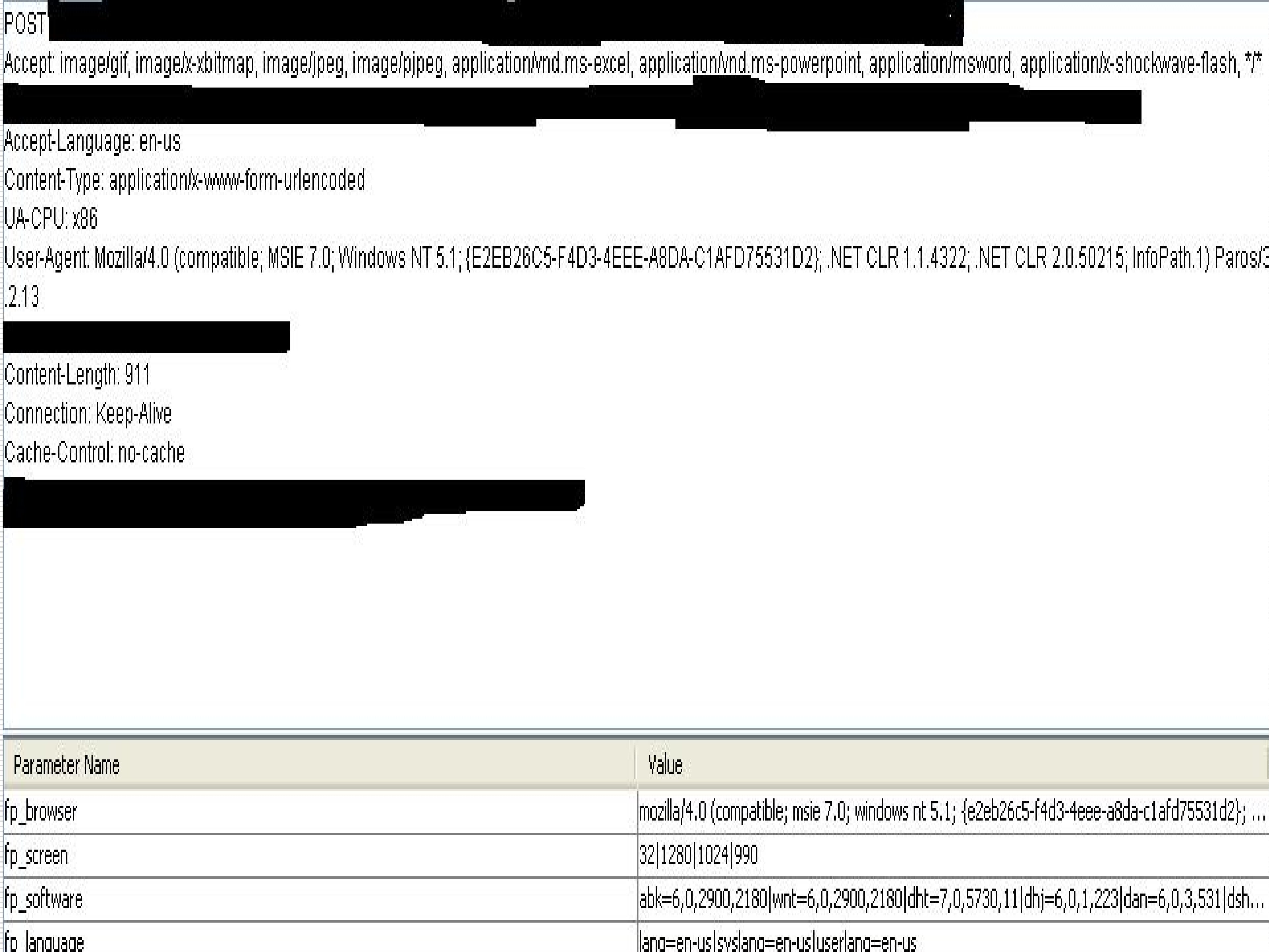
Device Fingerprinting

- How are 2 different servers with different SSL sessions keeping state?
- Analyze Post body
  - What are they trying to do?
  - How are they doing it?
  - Dissecting parameters and values



```
POST https[redacted]
Host:[redacted]
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.8.0.10) Gecko/20070216 Firefox/1.5.0.10 Paros/3.2.13
Accept: text/xml,application/xml,application/xhtml+xml,text/html;q=0.9,text/plain;q=0.8,image/png,*/*;q=0.5
Accept-Language: en-us,en;q=0.5
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
Connection: keep-alive
[redacted]
Cookie: [redacted]
Content-Type: application/x-www-form-urlencoded
Content-Length: 9526
```

Parameter Name	Value
[redacted]	[redacted]
[redacted]	[redacted]
[redacted]	[redacted]
txtUserID	[redacted]
btnValidateSignon	Continue
fp_browser	
fp_screen	
fp_software	
fp_timezone	
fp_language	
pm_fp	version=1&pm_fpua=mozilla/5.0 (windows; u; windows nt 5.1; en-us; rv:1.8.0.10) gecko/200...
TestJavaScript	OK



```
POST [redacted]
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, application/vnd.ms-excel, application/vnd.ms-powerpoint, application/msword, application/x-shockwave-flash, */*
[redacted]
Accept-Language: en-us
Content-Type: application/x-www-form-urlencoded
UA-CPU: x86
User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 5.1; {E2EB26C5-F4D3-4EEE-A8DA-C1AFD75531D2}; .NET CLR 1.1.4322; .NET CLR 2.0.50215; InfoPath.1) Paros/3
.2.13
[redacted]
Content-Length: 911
Connection: Keep-Alive
Cache-Control: no-cache
[redacted]
```

Parameter Name	Value
fp_browser	mozilla/4.0 (compatible; msie 7.0; windows nt 5.1; {e2eb26c5-f4d3-4eee-a8da-c1afd75531d2}; ...
fp_screen	32|1280|1024|990
fp_software	abk=6,0,2900,2180|wnt=6,0,2900,2180|dht=7,0,5730,11|dhj=6,0,1,223|dan=6,0,3,531|dsh...
fp_language	lang=en-us|syslang=en-us|userlang=en-us

Device Fingerprinting - Analysis

pm_fpua = mozilla/5.0 (windows; u; windows nt 5.1; en-us; rv:1.8.0.10) gecko/20070216 firefox/1.5.0.10|5.0 (Windows; en-US)|Win32

pm_fpsc = 32|1024|768|768

pm_fpsw = def|pdf|swf|qt6|qt5|qt4|qt3|qt2|qt1|j11|j12|j13|j14|j32|wpm|drn|drm

pm_fptz = -4

pm_fpln = lang=en-US|syslang=|userlang=

pm_fpjv = 1

pm_fpco = 1

Device Fingerprinting - Analysis

```
auth_deviceSignature         "appCodeName":"Mozilla",

"appName":"Microsoft Internet Explorer","appMinorVersion":"0",

"cpuClass":"x86","platform":"Win32","systemLanguage":"en-us",
"userLanguage":"en-us",

"appVersion":"4.0 (compatible; MSIE 7.0; ..UA Stuff..)",

"userAgent":"Mozilla/4.0 (compatible; ..More UA Stuff..)",

"plugins":[{"name":"Adobe Acrobat Plugin","version":"1"},
{"name":"QuickTime Plug-in","version":".."},
{"name":"Windows Media Player Plug-in Dynamic Link Library","version":""},
{"name":"Macromedia Shockwave Flash","version":"8"},
{"name":"Java Virtual Machine","version":""}],
"screen":{"availHeight":990,"availWidth":1251,"colorDepth":32,"height":1024,"
      width":1280},
```

Device Fingerprinting - Analysis

- The application is trying to gather information specific to your device to form a fingerprint
- How can their web server interrogate you device?
  - Javascript of course!
- Reverse Engineering isn’t hard when you have source code…

Device Fingerprinting - Analysis

```
/*  This function captures the User Agent String from the
    Client Browser    */
    function fingerprint_browser ()
    {

/*  This function captures the Client's Screen Information */
    function fingerprint_display ()
    {
```

That wasn’t too hard

Device Fingerprinting

Failing Device Fingerprinting

- Challenge questions
- One time password
  - Out of band delivery
  - Session ID is not enforced (usually)
- Successful Authentication
  - Picture and pass phrase for mutual auth
  - Persistent cookie is set (Are you using a private or public computer?)

Device Fingerprinting - Attack

- Fuzz fingerprinting parameters
  - Determine failure thresholds
  - Site specific
  - IP lookup
- Challenge Questions
  - Lack of randomization
  - Q1, Q2, Q3, Q1 …
  - Trivial to enumerate valid usernames

Device Fingerprinting - Attack

- Multiple servers and redirects
  - The client keeps state
  - You are the client
- Systems that use a single session
  - Out of state requests are possible
  - Force an OTP to be sent
  - Force challenge questions

Device Fingerprinting - Attack

- Mutual Authentication
  - Picture and Passphrase
  - Servers mask Get request through GUIDs or Stream Ciphers

How can we defeat this?

1. IV Collision (exhaustive requests)
2. MitM On the Fly replacement
3. Clear text Alt tags

Device Fingerprinting - Attack

All Implementations of this System have the same Alt tag for each unique image.

- □ Shared catalog of images
- □ Having access to any one app using this system allows you to mirror the image catalog
- □ No need to attack the app’s dynamic link function

Device Fingerprinting – Measure Up

Designed to Combat

- Phishing
- Transaction Fraud
- Identity Theft

Device Fingerprinting – Phishing

- Phishing is targeted at a specific organization
- Attacker can simply copy the fingerprinting Jscript from target site
- As long as username is correct, failing fprint will present challenge questions
- Attacker gets answer, and the questions are not random

Device Fingerprinting – Phishing

- Spear-phishing easier than ever
  - Valid account names can be enumerated
  - Device fingerprint can be brute forced
  - What are the chances valid account names are used for email? (user@yahoo, user@hotmail, user@aol, etc.)
  - A phishing email including a user’s security image and passphrase has a greater chance of success

Device Fingerprinting - Fraud

- Does absolutely nothing to stop Fraud
  - Inheritance trust model still applies
  - Once authenticated, all transactions are valid
- Identity Theft
  - Datamasking (account #********1234)
  - Check Images > just an account number
  - E-Statements or Tax forms

One Time Passwords

- Covered some of this already
  - Only delivery is out of band
- Hardware and “Soft” tokens
  - If the app isn’t enforcing all phases within a single session, same issues apply
  - Long or non-existent TTLs
- OTPs are most effective when required for every login

One Time Passwords

- Can be Man in the Middle’d
- Email or SMS delivery sets a pattern for the user
- XSRF is possible in conjunction with a phishing site

One Time Passwords – Measure Up

- Better than fingerprinting because its more difficult to be transparent
- Trains the user to trust email more
  - Clicking links
  - Using email for security purposes
- Does nothing to combat Fraud or Identity Theft
  - Inheritance trust model still applies

Knowledge Base Archives

- Not nearly as common (but out there)
- Used in conjunction with persistent cookie (usually)
  - By definition, public records are used
  - “Skip this question” option
- Randomization works in our favor
  - Multiple requests from multiple sessions
  - Pattern analysis

Knowledge Base Archives

- In 2002 did you buy
  1. Honda Accord
  2. Toyota Camry
  3. Ford Taurus
  4. None of the Above
- In 2002 did you buy
  1. Nissan Sentra
  2. Chevy Cavalier
  3. Ford Taurus
  4. None of the Above

Knowledge Base Archives

- Less effective than challenge questions
  - Can be defeated through response analysis with zero prior knowledge
- Same shortcomings as other solutions
  - Doesn’t stop phishing
  - Doesn’t stop transaction fraud
  - May make Identity Theft easier

Is There a Better Way?

- Mutual Auth
  - Responses must always be given
  - Same response must always be given for same authentication criteria
  - Auth should be algorithmic
- Challenge Questions
  - Still single factor
  - Replacing something the user knows with 2 things the user knows
  - Flawed by design – users can pick simple questions with simple answers

Is There a Better Way?

- Device Fingerprinting
  - Current implementations can be bypassed or replicated with ease
  - Replacing something the user knows with something the computer knows
  - Forgiving thresholds and persistent cookies aren’t buying us anything

Is There a Better Way?

- Stop fingerprinting devices, start fingerprinting behaviors
  - True transaction based behavior analysis and anomaly detection
  - HTTP header information != behavioral analysis
- Hurdles for secure implementation
  - Sheer volume of data
  - Bolt On vs. Built In – this needs to be built into the application itself

Is There a Better Way?

- Use a Positive Authentication Model
  - New transactions should require strong auth
  - Use hash values of transactions to prevent tampering
    - Trojans and BHOs that target specific institutions are not uncommon
    - Sit and Wait – on the fly transaction replacement by malware is in the wild
  - Force the user to review and verify login events and transactions
    - Make the user be involved in the security of their account

Is There a Better Way?

- Hardware tokens have a good security record
  - If the company doesn’t want to pay, let the user opt-in and share the cost

Conclusions

- Why did I do this?
- Traditional attack vectors are still a threat
  - This does not address any other vulnerability types, which are still an issue
  - If XSS exists, these controls are generally worthless (persistent cookie)
  - Browser based vulnerabilities are still a problem
- Putting controls in the wrong place – too much attack surface

Conclusions

- Financial Industry Problems
  - If a customer loses their checkbook or credit card, the FI picks up the tab
  - Who pays for online fraud due to phishing or malware?
  - Lose/Lose
    - Company – Free online services may go away (Risk vs Reward)
    - Customer – Stop using online systems, because they’re covered in the physical world

Conclusions

- The Cycle
  - People complain about phishing, fraud, and ID theft
  - Government regulates and legislates
  - Private sector implements technology that satisfies legal requirements but does not address the real problem
  - Attackers adapt
  - Rinse, Repeat

Conclusions

Why we’re worse off

- □ False sense of security to end user
- □ Taking a step backwards in some cases
- □ Most technologies being deployed aren’t addressing the real problem
- □ App vendors need to build it in, not bolt it on
- □ Security products should reduce attack surface, not increase it

Thank You