MAXIMUM SECURITY CFML

Pete Freitag, Foundeo Inc

WHO AM I?

- Owner of Foundeo Inc (visit our booth)
    - ColdFusion Products - FuseGuard
    - ColdFusion Services - HackMyCF.com
    - ColdFusion Consulting - Security Audits, development, etc.

AGENDA

- Security Principles & Challenges
- Loggings, Auditing, Intrusion Detection
- Authentication, Authorization, Hashing & Salting
- Specific Vulnerabilities
    - How to hack them
    - How to fix them

MORE SECURITY AT CFOBJECTIVE

- FuseGuard Demo - AdHoc Room Tomorrow at 1:45
- BOF - Locking down ColdFusion Servers Tomorrow Night @8
- Advanced Web Application Security - Jason Dean Tomorrow at 11:30
- Application Intrusion, Detecting & Tracking - Dave Ferguson Saturday 10:15

SECURITY - IT'S IMPORTANT

- Ignoring Security Can Lead To:
  - Embarrassment
  - Loss of Customers
  - Loss of Job or Lawsuits
  - Loss of Hair

SECURITY BREACHES ARE EXPENSIVE

- 2010 Cost of a Data Breach
  - $214 Per Record / Customer
  - $7.2 Million Average Total Cost per Incident

Source: http://www.symantec.com/about/news/resources/press_kits/detail.jsp?pkid=ponemon

WHAT IS THE MOST COMMON THREAT?

- Negligence, causing 41% of breaches.

Source: http://bit.ly/i6SPng

ISN'T SECURITY GREAT?

photo credit: mobileedgelaptopbags on flickr

YOU WOULD PROBABLY RATHER THINK ABOUT:

$(document).ready(function() {});

oo.goodness()

orm

caching

performance

jQueryMobile

to the cloud baby!

clusters

android

pixels

iphone

tablets

mobile version

START THINKING ABOUT SECURITY

- Keep security in mind when writing code, and architecting software.
  - This goes a long way towards making your applications more secure.
  - When security is an afterthought it is much more difficult to deal with.

YOUR BOSS THINKS ABOUT

how much?

does it make money?

done yet?

blackberry version?

SECURITY FOR MANAGERS

- Most managers or clients have no idea how to secure a web app
    - They are trusting that you know what you are doing.
- Security is an invisible feature, not all clients / managers want to invest in.

HACKERS LOVE WEB APPS

- Easy to produce insecure code
- Easy to Hack (PhD in Cryptography not necessary!)
- Easy to find vulnerable sites (google)

WHY DO HACKERS HACK?

- Spy Stuff & Money not the only reason...
  - Server Control (to use for other attacks, spam, etc)
  - Defacement
  - Illogical Reasons, they're criminals and probably a few cards short of a full deck.

HAVE YOU BEEN HACKED?

WHAT GOES THROUGH YOUR HEAD WHEN HACKED:

- What did they do?
- What was compromised?
- How did they do it?
- You need some evidence

LOGGING

- Log anything that might be useful evidence:
  - Exceptions
  - File Operations
  - Etc...
- Use CFLog, database, log4j, syslog, etc...

AUDITING

- Logging of Successful or Failed Events, for example:
    - Logins
    - Add / Edit / Delete Data
        - In some cases Read events
    - Other business specific events

INTRUSION DETECTION

- Intrusion Detection Software can help block or log malicious requests to your application.
  - Web Application Firewalls

AUTHENTICATION & AUTHORIZATION

- Almost Every Application Must Authenticate (login) and Authorize (permissions) Users
    - Lots of room for error
    - Many common mistakes
    - Session Hijacking

INSECURE PASSWORD STORAGE

- Passwords should be stored hashed using a secure cryptographic hashing algorithm and salted.
  - What's a hashing algorithm?
  - Salt?

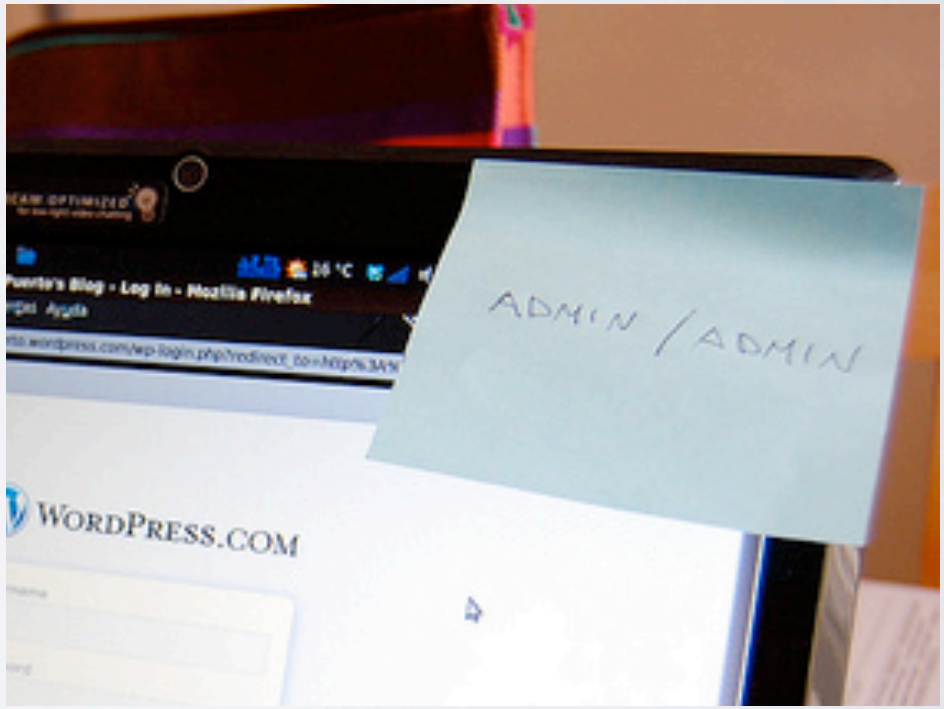

photo cc: http://www.flickr.com/photos/reidrac/4696900602/

HASHING

- A hash provides a **one way** encoding of a string into a fixed length string.
  - Unlike Encryption which is two way (you can get the original string again if you have the key)
- Use ColdFusion's **Hash(string, algorithm, encoding)** function:
  - Hash("password", "SHA-512")

HASH ALGORITHMS

- MD5 - Default Algorithm of the Hash Function, Fast not as secure
- SHA - Secure Hash Algorithm FIPS
  - SHA-1 160 bit Algorithm designed by the NSA
  - SHA-2 (SHA-256 and SHA-512) also designed by the NSA
  - SHA-3 winner will be announced by NIST in 2012
- Algorithm support determined by JCE. ColdFusion Enterprise installs RSA BSafe Crypto-J Provider for FIPS-140 Compliance.

HASH EXAMPLES

Hash("password", "MD5"):
5F4DCC3B5AA765D61D8327DEB882CF99

Hash("password", "SHA1"):
5BAA61E4C9B93F3F0682250B6CF8331B7EE68FD8

Hash("password", "SHA-512"):
B109F3BBBC244EB82441917ED06D618B9008DD09B3
BEFD1B5E07394C706A8BB980B1D7785E5976EC049B
46DF5F1326AF5A2EA6D103FD07C95385FFAB0CACB
C86

SALT THAT HASH

- Hashing a string always results in the same hash string.
  - If an attacker gains access to your hashed passwords, they can use a Rainbow Table to reverse the hash.
  - So you need to salt it.

photo cc: http://www.flickr.com/photos/tudor/163906410/

EACH USER HAS SAME PASSWORD

uid	password
1	5F4DCC3B5AA765D61D8327DEB882CF99
2	5F4DCC3B5AA765D61D8327DEB882CF99
3	5F4DCC3B5AA765D61D8327DEB882CF99

← No Salt

uid	password
1	8FD974D2D58F875F968AF667994C951B
2	DF982CE25D47C6E8ECA7BEE61AE972C3
3	BE721CAA292A226EA58E8089CF422407

← Salted

SALT

- Each user should have a **unique** salt string associated with it.
  - GenerateSecretKey("AES")
  - Rand("SHA1PRNG")
- Hash(form.password & saltString, "SHA-512")

HASH + SALT EXAMPLE

```
<cfquery name="user">
    SELECT password, salt
    FROM users
    WHERE username = <cfqueryparam value="#form.username#"
                                   cfsqltype="cf_sql_varchar">
</cfquery>
<cfif user.password IS Hash(form.password & user.salt, "SHA-512")>
    <!--- authenticated --->
</cfif>
```

SECURE AUTHENTICATION TIPS

- Account Lockout
  - Use your audit log to determine if an unusual amount of failed logins are taking place.
- Sleep on failed logins (be sure not to DOS yourself)
- Password Strength Requirements
  - No need to limit max length of password

SECURE AUTHENTICATION TIPS

- Hash Iterations - Put your hash in a loop to increase execution time to deter brute force.
    - Beware of timing attacks

SESSION HIJACKING

If I know your
CFID & CFTOKEN
(or JSESSIONID)
then I can impersonate you.

SECURE AUTHENTICATION TIPS

- Never pass session id's in the URL
  - Always set addtoken=false when using the cflocation tag.
- Require SSL and Secure Cookies
- Set HTTPOnly Flag on cookies (see my blog for example code)
  - CF 9.0.1 Added JVM Argument
    - -Dcoldfusion.sessioncookie.httponly=true

INSECURE FILE UPLOAD

- **Very Common, Very Dangerous**
- **The Risk:**
  - An attacker uploads and executes a file on your server.

VULNERABLE CODE

```
<cffile action="upload"
        filefield="photo"
        accept="image/gif,image/jpeg,image/png"
        destination="#ExpandPath("./photos/")#">
```

WAIT A SEC...

- Doesn't the **accept** attribute limit the types of files that can be uploaded?

DEMO

UH-OH!

- The mime type used by the **accept** attribute is supplied by the client (from the web browser).

EXPLOITING IT

```
<cfhttp url="http://example.com/upload.cfm" method="post">
    <cfhttpparam file="#ExpandPath("code.cfm")#"
                 mimetype="image/gif"
                 type="file"
                 name="photo">
</cfhttp>
```

WHAT DID WE LEARN?

STILL VULNERABLE?

```
<cffile action="upload"
             filefield="photo"
             accept="image/gif,image/jpeg,image/png"
             destination="#ExpandPath("./photos/")#">

<cfif NOT ListFindNoCase("gif,jpg,png", cffile.ServerFileExt)>
    <cffile action="delete"
file="#cffile.ServerDirectory#/#cffile.ServerFile#">
<cfelse>
    <p>File Was Uploaded.</p>
</cfif>
```

DEMO

YES, STILL VULNERABLE

- Notice that the destination of cffile was **under the web root**.
    - The file was uploaded to the web root and may be executed before it is deleted milliseconds later.

Hacker

POST /upload.cfm

GET /photos/photo.cfm

Hacker uses a load tool to make repeated concurrent requests.

After a while, the attacker will get lucky

FIXING FILE UPLOADS

- Use but don't rely on the accept attribute.
- Always validate file extension
- Never upload under the web root.
    - Only copy files there once validated.
- Try / Catch & Delete

WHITELIST VS BLACKLIST

- Prefer whitelists over blacklists
  - eg: allow jpg, png, gif, pdf
- Black lists are very hard to maintain
  - eg: block cfm,cfc,jsp
  - Oops you missed: cfml, cfr, jws
  - Admin just installed php...

FILE UPLOAD TIPS

- Validate File Content if possible
    - IsImageFile(*path*)
    - IsPDFFile(*path*)
    - IsSpreadsheetFile(*path*)
    - jHOVE - Java API for additional types

FILE UPLOAD TIPS

- Deny execution for upload destination directory.
    - On Web Server
    - In ColdFusion (with Sandbox Security)
- Serve files from a static content server
    - Build your own
    - Amazon S3, etc.

FILE UPLOAD TIPS

- Set **mode** attribute of cffile on unix
  - eg: 640 = rw-r-----
    - 7 = read, write, execute (rwx)
    - 6 = rw
    - 4 = r
    - 0= no privledges

SQL INJECTION

- **The Risk:**
  - Attacker can run arbitrary SQL against your database.
  - Typically execute system commands on the database server.

VULNERABLE CODE

```
<cfquery datasource="#application.ds#" name="news">
    SELECT id, title, story
    FROM news
    WHERE id = #url.id#
</cfquery>
```

EXPLOITING IT

- Instead of news.cfm?id=1
- Attacker Runs:
  - news.cfm?id=1;DROP+Users
  - news.cfm?id=1+UNION+SELECT+...
- Quick Demo

FIXING SQL INJECTION

- Use the **<cfqueryparam>** tag for variables in a query when possible.

```
<cfquery datasource="#application.ds#" name="news">
    SELECT id, title, story
    FROM news
    WHERE id = <cfqueryparam value="#url.id#"
                    cfsqltype="cf_sql_integer">
</cfquery>
```

HOW CFQUERYPARAM WORKS

This:

```
<cfquery datasource="#application.ds#" name="news">
SELECT id, title, story FROM news
WHERE id = <cfqueryparam value="#url.id#" cfsqltype="cf_sql_integer">
AND category = <cfqueryparam value="#url.cat#" cfsqltype="cf_sql_integer">
</cfquery>
```

Is sent to the DB as:

```
SELECT id, title, story FROM news
WHERE id = ?
AND category = ?
```

data[1] = 123 ⟵ ID

data[2] = 913 ⟵ CAT

CFQUERYPARAM

- Works in WHERE clauses, INSERT values, and UPDATE values.
    - Some places it does not work SELECT TOP *n*, ORDER BY (depends on DB)
- Can be used with lists in an IN statement using list=true

WHEN YOU CAN'T USE CFQUERYPARAM

- Be sure you have validated the variable as a simple type.
    - EG: SELECT TOP #Val(url.top)#
        - Val() returns 0 when given non-numeric input

IS SQL INJECTION **POSSIBLE WITH** COLDFUSION **ORM?**

SQL INJECTION - ORM

- SQL Injection is possible when writing HQL queries:

```
ORMExecuteQuery("FROM Entity WHERE id = #url.id#")
```

- Easily Prevented:

```
ORMExecuteQuery("FROM Entity WHERE id = :id", {id=url.id})
```

CROSS SITE SCRIPTING

- **The Risk:**
  - Session Hijacking
  - Phishing

VULNERABLE CODE

```
<cfoutput>
  Hello #url.name#
</cfoutput>
```

EXPLOITING XSS

- Instead of hello.cfm?name=pete
- Attacker runs:
  - hello.cfm?name=<script>alert('pete')</script>
- Demo's

FIXING XSS

- One Solution: Strip all harmful characters
    - < > ' " ( ) ; #
- Not always a realistic solution.

FIXING XSS

- Encode variables to escape special characters. (eg < becomes <)
    - The best way to do this depends on where the variable is output, in a tag attribute, inside JavaScript, etc.

OUTPUT CONTEXT'S

Context	Example
HTML	`<p>Hello #url.name#</p>`
HTML Attribute	`<div id="#url.name#" />`
JavaScript	`<a onclick="hi(#url.name#)" />` `<script>#var#</script>`
CSS	`<div style="font-family: #url.name#" />` `<style>#var#</style>`
URL	`<a href="hi.cfm?name=#url.name#" />`

HTML CONTEXT

- XMLFormat() or HTMLEditFormat()
  - XMLFormat Escapes < > ' "
  - HTMLEditFormat Escapes <> "

USING ESAPI

- OWASP Enterprise Security API
  - Java API that has encoder methods for each context.
  - http://code.google.com/p/owasp-esapi-java/
  - ColdFusion Security Hotfix (CF8-9) APSB11-04 includes ESAPI jar files!

USING ESAPI

Context	Method
HTML	`esapi.encodeForHTML(variable)`
HTML Attribute	`esapi.encodeForHTMLAttribute(variable)`
JavaScript	`esapi.encodeForJavaScript(variable)`
CSS	`esapi.encodeForCSS(variable)`
URL	`esapi.encodeForURL(variable)`

```
<cfset esapi = CreateObject("java", "org.owasp.esapi.ESAPI").encoder()>
```

WHAT IF MY USER MUST SUBMIT HTML?

- You need to make sure the html is valid
- Does not contain any script, iframe, object, style, etc tags.
- HTML attributes do not have harmful JS event handlers or exploit CSS hacks in the style attribute.
- Very difficult to write something like this

ACCEPTING HTML

- AntiSamy for Java
  - Create a policy defining allowed HTML
  - ESAPI has integrated AntiSamy in its Validator implementation
    - ESAPI.validator().isValidSafeHTML()
  - Ask me who's "Samy" later.

WHY NOT SCRIPTPROTECT?

- ColdFusion 7 Added ScriptProtect feature to "protect" form, url, cgi and cookie variables from XSS
- Very limited and easy to bypass.
- Not a solution to XSS, but feel free to enable it.

PATH TRAVERSALS

- **The Risk:**
  - Allows attacker to read any file CF has permission to read.

VULNERABLE CODE

```
<cfinclude template="files/#url.page#">
```

EXPLOITING IT

- Instead of page.cfm?path=about.cfm
- Attacker runs:
  - page.cfm?path=../../any/file/on/the/server
- Demo

VULNERABLE?

```
<cfinclude template="#url.page#.html">
```

Sure, but you can only include .html files right?

OH MY...

- page.cfm?page=../anything/i/want.cfm%00
  - Using a URL Encoded null byte %00 causes anything after it to be ignored by the internal File IO operations.

FIXING PATH TRAVERSALS

- **Applies to:** cfinclude, cfmodule, cffile, File Functions, any code that deals with file paths
- **To fix:**
  - Avoid using unsafe variables in code that deals with files.
  - If you must use a user supplied variable validate it.

SECURITY GUIDELINES

- Don't Trust Inputs
  - Validate All Inputs. More Validation = More Security
    - Fine grained validation is best.
    - Remember that the entire HTTP request is an input.

SECURITY GUIDELINES

- Careful What You Output
  - Scrub and Sanitize all outputs
    - Ensure that all variables are encoded and escaped properly

SECURITY GUIDELINES

- Bring Security into your Unit Tests
  - Ensure that your app does not accept malicious input
- Catch Exceptions with try/catch, onError, cferror
  - Don't disclose system details to end user, log the details

SECURITY TOOLS FOR COLDFUSION

- FuseGuard - Web Application Firewall For ColdFusion
- HackMyCF.com - ColdFusion Server Security Scanner

Questions?

Thank You

pete@foundeo.com | foundeo.com | petefreitag.com
@pfreitag