Role of Computer Security in Identity Theft (November 2007)

Charanjit Dhanoya, Chendursundaran Kumaragurubaran and Joey Ting

***Abstract*—Phishing is becoming a popular form of online identity theft. As an analysis this paper explores the techniques used to construct phishing attacks and also determine how anti-phishing tools work and their effectiveness it sets the footnote at the bottom of this column. More specifically, we want to know what techniques are used in these tools to detect or prevent phishing attacks.**

***Index Terms*—Dynamic Name Server, DNS, Man in the middle, MITM, Cross Site Scripting, XSS, Universal Resource Locator, URL**

I. Introduction

Identity Theft is a crime when a criminal uses victim's personal identifying information such as name, social insurance number to commit fraud. Phishing and Pharming are considered as online identity theft. They are distinguished from offline identity theft such as card skimming and "dumpster diving," as well as from large-scale data compromises in which information about many individuals is obtained at once. Phishing is typically carried out by email and often directs users to enter details at a website to acquire usernames, passwords and credit card details. Online websites like eBay, PayPal and online banks are common targets.

Phishing accounts to $1 billion a year in direct losses in the US. Indirect losses are much higher, including customer service expenses, account replacement costs, and higher expenses due to decreased use of online services in the face of widespread fear about the security of online financial transactions. Both the frequency of phishing attacks and their sophistication is increasing dramatically.

Manuscript received November 19, 2007.
C. Dhanoya (e-mail: charanjeet9@hotmail.com).
C.Kumaragurubaran (e-mail: daran9@gmail.com).
J. Ting (e-mail: touareg03@hotmail.com).

II. Techniques/TOOLS Used in Phishing Attacks

A. Overview

Phishing attacks rely upon a mix of technical deceit or technical subterfuge and social engineering practices [1]. These attacks are usually done to steal consumers' personal identity data and financial account credentials. Social engineering is creating fake emails to trick consumers and lead them to fraudulent websites and trick them into entering their personal and financial information like credit card numbers, bank account information, usernames and passwords of certain sites and so on. Technical subterfuge techniques install crime ware onto the user's computers and directly steal credentials often using Trojan key logger spywares. Phishing attacks take different forms and use different techniques. Attackers use a number of methods to trick the customer into doing something with their server and/or supplied page content. There are many different ways to do this.

Some phishing attack techniques used are as follows [2]:

1) Email and Spam.
2) Web-based Delivery.
3) IRC and Instant Messaging.
4) Trojaned Hosts
5) Man-in-the-Middle Attack
6) URL Obfuscation Attacks
7) Cross-site Scripting Attacks
8) Preset Session Attack
9) Hidden Attacks
10) Observing Customer Data

Out of the above mentioned attacks, attacks using emails and spams are the most common. In this report, we will limit to Man-in-the-Middle Attack, URL Obfuscation and Cross-site Scripting attacks.

B. Man-in-the-Middle Attack

Man-in-the-middle (MITM) attack is one of the most successful types of attacks. In this type of attack, the attacker places himself between the user and the real web-based application. This attack works for both HTTP and HTTPS communications. Figure 1 shows the structure of MITM attack.

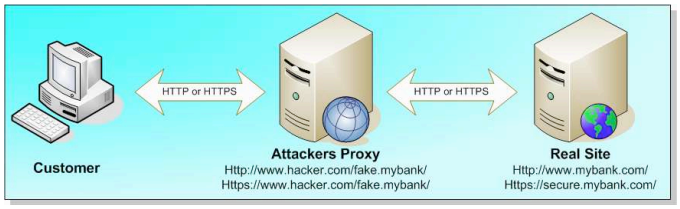

Fig. 1. Man-in-the-middle attack structure.

The only main concern of the attacker to make the attack successful is to be able to direct the users toward his proxy server. This can be done in various ways.

Some are listed below [2]:

1) Transparent Proxies
2) DNS Cache Poisoning
3) URL Obfuscation
4) Browser Proxy Configuration

With respect to identity theft, Transparent Proxies and DNS Cache Poisoning are the most common ones among attackers[3].

Transparent Proxies

A 'transparent proxy' is a proxy that does not modify the request or response beyond what is required for proxy authentication and identification. One does not have to configure it for HTTP traffic. As they are built as part of network architecture, all port 80 traffic flows through them. These are situated on the same network segment or located on route to the real server (e.g. corporate gateway or intermediary ISP) and it can intercept all data by forcing all outbound HTTP and HTTPS traffic through itself[2].

DNS Cache Poisoning

DNS Cache Poisoning attack is based on simple convention of IP to host resolution. DNS servers are constantly sending out questions about IP addresses to hosts and receiving its answers. These servers do not authenticate the source of the answers i.e. authentication of the origin. The simplest form of cache poisoning is simply sending fake answers to someone's DNS server. There are steps to avoid it; however, sometimes they are compromised. The attacker in MITM attack uses DNS Cache Poisoning to disrupt normal traffic routing by injecting false IP addresses for key domain names. For example, the attacker poisons the DNS cache of a network firewall so that all traffic destined for the My Bank IP address now resolves to the attackers proxy server IP address [2].

C. URL Obfuscation

This attack is very easy to execute and has been used by attackers when phishing was in its early stages. This attack is basically making the users follow a link to the attacker's fake website without the user realizing it. There are different methods the attacker uses to disguise the URL.

Some modifications the attacker makes to trick users are listed below [3]:

1) Using Strings
2) Using @ sign
3) URL Encoding

Using Strings

This uses a familiar and related text string within the URL. Example:http://XX.XX.43.102/ebay/account_update/now.php. This in a real time scenario will point towards a web server hosting a fake login screen for your Ebay account.

Using @ sign

When @ sign is used in the URL, the content on the left side of @ sign is ignored and the domain name or IP address on the right side of @ sign is treated as the actual domain. Example: http://www.citybank.com/update.pl@xx.xx.43.102/usb/upd.pl.

URL Encoding

In this method, the URL or portions of the URL is encoded to disguise its true value using hex, dword, or octal encoding. It is usually combined with @ which can also be disguised. Example:http://www.visa.com@%32%32%30%2E%36%38%2E%32%31%34%2E%32%31%33, which translates into 220.68.214.213

Some other tricks that are used are Bad domain names Friendly login URL's, Third-party shortened URL's, Host name obfuscation, URL as button, URL Redirection, Double Redirect.

D. Cross-Site Scripting Attacks

Cross-site scripting (XSS) attacks occur when an attacker uses a web application to send malicious code, generally in the form of a browser side script, to a different end user" 4. CSS techniques are usually because of poor web application development processes. Using this method, the attacker steals client cookies, or other sensitive information, which identifies the client with the web site. Using this unique information of the client, he/she interacts with the site specifically, impersonating the user.

The various formats for XSS injection into valid URL's are [2]:

1) Full HTML substitution
2) Inline embedding of scripting content
3) Forcing the page to load external scripting code

Full HTML substitution

http://mybank.com/ebanking?URL=http://evilsite.com/phishing/fakepage.htm

Inline embedding of scripting content

http://mybank.com/ebanking?page=1&client=<SCRIPT>evilcode...

Forcing the page to load external scripting code

http://mybank.com/ebanking?page=1&response=evilsite.com%21evilcode.js&go=2

Following is an example of the working of a cross-site scripting attack.

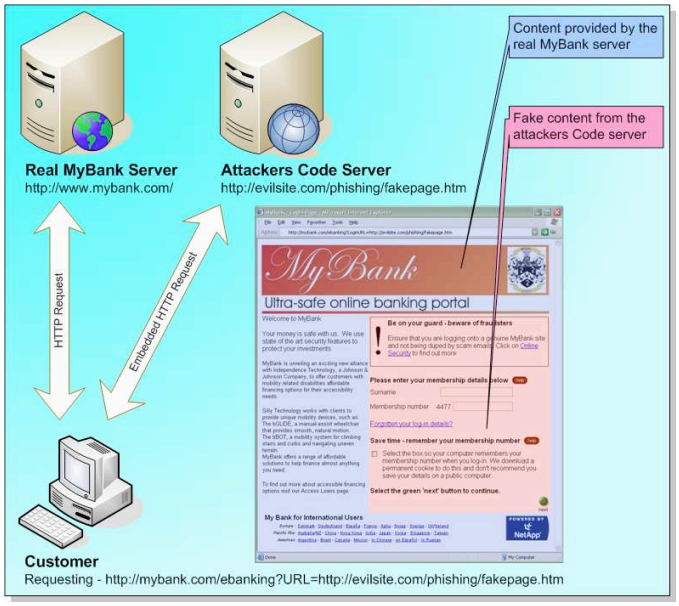

Fig. 2. Cross-site scripting attacks.

The customer receives the following URL via an email sent by the attacker: http://mybank.com/ebanking?URL=http://evilsite.com/phishing/fakepage.htm

Here the customer is directed and gets connected to the real MyBank web application. However, the bank website has a poor application coding. The ebanking component accepts an arbitrary URL for insertion within the URL field of the returned page. Now, the attacker has the control and rather than the banks' original authentication form embedded within the page being shown, the attacker references a page under control on an external server (http://evilsite.com/phishing/fakepage.htm). Here, the user is completely unaware of the fake authentication page. The URL in the example may appear obvious and can be detected; however, the attacker can easily obfuscate it using the URL obfuscation techniques explained earlier2. For example, http://evilsite.com/phishing/fakepage.htm may instead become: http%3A%2F%2F3515261219%2Fphishing%C0%AEfakepage%2Ehtm

Cross-Scripting attacks are one of the most powerful phishing attacks. With the help of XSS attacks, an attacker can even gain access to the shell of a restricted web server[3].

III. Techniques/Tools Used in Phishing Prevention

There are three main pillars of anti-phishing: preventing, detection, and notification.

A. *Prevention (websites)*

Customize login screen for each user

To provide verification to the customer that the website he/she is logging into is legitimate, the website can display some information that is unique to the user, before the user provides any sensitive information [5]. A good example of this is online banking section of ING Direct Canada's website. To log in, the customer first provides the account number. The website then displays a photo and a phrase that the customer has chosen before. If the photo and phrase displayed matches what the user has chosen before, then the customer can be reasonable certain that the website is legitimate, and can then enter in her password to gain access.

This process does not require a significant investment by the company to implement, and is simple for the customer to use. By allowing customers to choose their own photo and phrase, they will be more likely to remember what their photo and phrase is, meaning that it will be more obvious when the wrong photo and/or phrase is displayed. The disadvantage of the process is that it requires the user to check the photo and the phrase for every login attempt, so there is a significant chance that the user will eventually ignore the photo and the phrase during login, because they may simply stopped checking the photo and phrase.

One-time password

In addition to the username/account number and the password usually required to login, a one-time password can be used as well. The one-time password has a limited validity period (usually 60 seconds or less), making the one-time password useless if it is stolen, unless the attacker attempts to use it before the validity period expires [5]. The one-time password is usually generated to the user via a portable hardware device, commonly known as a token. The token has a display which shows the current one-time password, and the display is updated as the one-time password expires and the next password becomes valid. An example of this would be RSA's SecureID tokens, commonly used in corporate VPN access.

To the attacker, one-time passwords limit the usefulness of a user's username/account number and regular password, as three pieces of information is now needed for a successful login, and the one-time password is constantly changing. Also, a careless user cannot bypass the use of the one-time password, meaning that this anti-phishing measure is always used and cannot be forgotten. A downside to one-time passwords is that an emerging class of phishing attacks, called "real-time man-in-the-middle", aims to use the one-time password before it expires. Also, users must have the hardware token with them to login, which suggests that users should carry the token with them, increasing the chances that the token would be lost or stolen. Occasionally, tokens may malfunction, and this may require the token to be sent back to the company for repairs.

Multiple passwords

Multiple passwords can be used in an online transaction [5]. For example, in an online banking environment, one password can be used to login, but a special 'transaction' password must be used whenever money is to be transferred out of an account in some way. This method is used by some online investing websites, such as TD Waterhouse's Webbroker.

This method depends on the user being able to verify the integrity of the account information upon logging in, but before a transaction takes place. If the user sees that the

account information displayed does not seem to be valid, then the user should not proceed to perform any transactions. However, it can be argued that the attacker can still quite easily obtain the login password, and that this already exposes too much information to the attacker, even if the attacker is not able to perform any transactions.

Website watermarking

It is usually feasible for an attacker to simply make a copy of the webpage code and modify it to run on a different server [6]. To guard against this, webpage code can be watermarked. For example, a script can be embedded into a webpage that checks to see if the host server's URL or domain matches the legitimate URL or domain [7]. If not, the script can send a message to the legitimate website, at which point the webmaster can take action. Visual watermarking of images, clearly showing the URL or the legitimate website, is also possible, but is unlikely to catch the attention of the user. Also, as an attacker can simply copy the images themselves and use it on the phishing site, it is clear that watermarking images provides no real help.

B. Detection

Websites blacklists and whitelists

A technique used in many anti-phishing software, a phishing black-list is simply a list of URLs that are known to be addresses of phishing websites. Whenever a user's web browser navigates to a different URL, the anti-phishing software checks the new URL against the whitelist (usually stored locally) and blacklist (usually stored on a server on the Internet). If the URL is found on the blacklist, then the anti-phishing software takes action to block the loading of the website and to notify the user. If the URL is on the whitelist, it will be loaded, regardless of if it's also on the blacklist or not. An obvious downside to blacklists is that they need to be kept up-to-date to be effective. Also, if a user visits a URL that is not listed in the blacklist, this does not mean that the website is legitimate. In most web browsers with anti-phishing features, this is the primary way that they detect phishing websites.

Website heuristics

Heuristics are also used in detecting phishing websites; anti-phishing systems that use heuristics analyzes certain aspects of a website, such as the phishing website attack techniques discussed earlier in this report [8]. For example, heuristics can check for URL obfuscation and whether images are being loaded from the same domain as the webpage. The main advantage is that heuristics guards against new, unknown phishing websites. However, any website suspected by heuristics may or may not be legitimately a phishing website, whereas websites on a blacklist are confirmed to be phishing websites. A workaround to this problems is to let the user decide if he/she would like to view a webpage that is suspected (but not confirmed) to be a phishing website, but this requires the user to determine for his or herself if a websites is legitimate or not, and this somewhat defeats the purpose of using anti-phishing software.

Spam email filtering

Although spam email and phishing emails have their differences, today's spam filters are quite adept at filtering out phishing emails. Phishing emails attempt to imitate a legitimate organization to steal authentication information, while spam emails do not.

A common technique used to identify spam is content-filtering. This technique analyzes the contents of an email, searches for keywords and phrases, and gives the email a 'spam probability score' depending on the number and type of keywords and phrases it finds [9]. However, as phishing emails attempts to appear as if it is a legitimate email, it is much more likely to fool a content-filter system compared to spam emails. However, many current spam filters also employ heuristics, in addition to content-filtering, to detect spam email, and this combination is relatively effective at filtering our phishing emails [10]. Using heuristics means to look at the headers and the encoding of an email for signs of a spam email. For example, this technique will likely be suspicious of an email that contain a hyperlink that is linked to a particular URL (such as www.fakebank.com) but the linked text appears as a different URL (such as www.realbank.com). Heuristics may also perform tests related to mail servers; for example, it may attempt to perform a reverse DNS lookup of the sender's email address domain [9].

C. Notification

Suppose that a phishing website has been detected by some system on a user's computer. This system needs to perform action(s) to prevent the user from accessing the website, and notify the user of what is happening. This is actually a crucial part of preventing phishing attacks, even though the techniques used here (to not initiate loading of a webpage and to inform the user of an issue) and not specific to anti-phishing systems. Unfortunately, many computer users cannot be expected to understand the often-cryptic security warnings they encounter while visiting websites, and tend to just accept the security warnings without thought [11].

To see what notifications are used, two web browsers with phishing filters (Firefox 2 and Microsoft Internet Explorer 7) and a standalone security suite (Norton Internet Security 2007) were installed on a test computer, and several known phishing websites from [12] were visited. When visiting a phishing website, Firefox 2 grays out the webpage display area of the browser, and pops up a message from the address bar with two main options: "get me out of here!" and "ignore this warning". The grayed-out area of the browser, combined with the atypical wording of the options in the message, better catches the user's eye compared to most other web browser security warning messages. The phishing filter in Norton Internet Security 2007 shows an obnoxious warning message in place of where the webpage would be displayed in the browser

window, as well as a red warning message on the web browser's toolbar. Internet Explorer 7's warning message looks very similar to all its other error messages, such as those for security certificate errors that are commonly ignored.

D. Anti-Phishing Tools

For the end-user detection of phishing websites, anti-phishing tools can be divided into three categories: web browsers with anti-phishing filters, anti-phishing toolbars, and standalone security software packages with anti-phishing features. The two most popular web browsers, Firefox 2 and Microsoft Internet Explorer 7, integrate phishing filters that use website whitelists and blacklists, but not heuristics [13] [14]. Anti-phishing toolbars may use whitelists and blacklists exclusively (such as Netcraft) [15] or combine this with heuristics (such as Google Toolbar for Firefox) [16]. Standalone software packages with anti-phishing features almost always include both techniques; examples include Norton Confidential and McAfee SiteAdvisor [17].

For detecting of phishing emails, the trend has been to modify current anti-spam systems to better detect phishing emails. Investigation of antispam systems falls outside the scope of this report.

REFERENCES

[1] Anti-Phishing Working Group (online). Available: http://www.antiphishing.org/

[2] G. Ollmann. The Phishing Guide – Understanding and Preventing Phishing Attacks (Online). Available: http://www.nextgenss.com/papers/NISR-WP-Phishing.pdf

[3] ContentVerification (Online). Available: http://www.contentverification.com/man-in-the-middle/index.html

[4] Cross Site Scripting - OWASP (Online). Available: http://www.owasp.org/index.php/Cross_Site_Scripting

[5] J. Varghese. (2006, August). Anti-Phishing Techniques - Protection Measures [Online]. Available: http://palisade.plynt.com/issues/2006Aug/phishing-protection/

[6] J. Varghese. (2006, September). Anti-Phishing Techniques - Detection Measures [Online]. Available: http://palisade.plynt.com/issues/2006Sep/phishing-detection/

[7] S. Anbalahan. (2007, March). Anti-phishing Measure: Palisade [Online]. Available: http://palisade.plynt.com/issues/2007Mar/quiz/?show=ans

[8] Y. Zhang, J. Hong, L. Cranor. CANTINA: A Content-Based Approach to Detecting Phishing Web Sites [Online]. Available: http://www2007.org/papers/paper557.pdf

[9] J. Zdziarski, W. Yang, and P. Judge. Approaches to Phishing Identification Using Match and Probabilistic Digital Fingerprinting Techniques [Online]. Available: http://zdziarski.com/papers/phishing.pdf

[10] I. Fette, N. Sadeh, and A. Tomasic. Learning to Detect Phishing Emails [Online]. Available: http://www.cs.cmu.edu/~sadeh/Publications/Small%20Selection/www07%20FINAL%20SUBMISSION.pdf

[11] P. Gutmann. Phishing Tips and Techniques: Tackle, Rigging, and How & When to Phish [Online]. Available: http://www.cs.auckland.ac.nz/~pgut001/pubs/phishing.pdf

[12] PhishTank | Join the fight against phishing [Online]. Available: http://www.phishtank.com/

[13] Firefox 2 Phishing Protection [Online]. Available: http://www.mozilla.com/en-US/firefox/phishing-protection/

[14] (2006, October 8). Phishing Filter: Help protect yourself from online scams [Online]. Available: http://www.microsoft.com/protect/products/yourself/phishingfilter.mspx

[15] Netcraft Anti-Phishing Toolbar [Online]. Available: http://toolbar.netcraft.com/

[16] Google Safe Browsing for Firefox [Online]. Available: http://www.google.com/tools/firefox/safebrowsing/

[17] S N. Rebbapragada. (2006, December 8). First Look: SiteAdvisor Plus vs. Norton Confidential. PC World. [Online]. Available: http://www.pcworld.com/businesscenter/article/128067/first_look_siteadvisor_plus_vs_norton_confidential_.html