Phish Phactors: Offensive and Defensive Strategies

HAL BERGHEL

School of Informatics and Internet Forensics Laboratory
University of Nevada, Las Vegas
USA

Department of Computer Science and Software Engineering
University of Canterbury
Christchurch
New Zealand

JAMES CARPINTER

Department of Computer Science and Software Engineering
University of Canterbury
Christchurch
New Zealand

JU-YEON JO

School of Informatics and Internet Forensics Laboratory
University of Nevada, Las Vegas
USA

Abstract

Phishing attacks attempt to fraudulently solicit sensitive information from a user by masquerading as a known trustworthy agent. They commonly use spoofed emails in association with fake websites in order to coerce a user into revealing personal financial data. Phishing is now a serious problem with criminals adopting the well-developed and well-known techniques to exploit Internet users with sophisticated attacks. Phishers are known to have successfully attacked an estimated 1.2 million users and stolen an estimated US$929 million in the twelve months to May 2005.

ADVANCES IN COMPUTERS, VOL. 70
ISSN: 0065-2458/DOI: 10.1016/S0065-2458(06)70005-5

This chapter aims to provide the current status of phishing attack techniques and defense methods. We first provide an overview of the fundamental phishing techniques for delivering a successful attack, such as bulk emailing, fake websites and detection avoidance using a variety of obfuscation techniques. We then survey more sophisticated methods that may deceive even knowledgeable and vigilant users. These techniques do not rely on naïve email users and simple websites, but use highly realistic fake websites, generic hacking techniques (such as DNS poisoning or cross site scripting) or actively exploit browser vulnerabilities. For example, a Man-In-The-Middle attack or the use of DNS poisoning can easily fool even an advanced user who may be aware of phishing attacks.

Quite a few defensive methods have been developed, although many are still in the early stage of development. URL obfuscation can be rather reliably detected using analysis algorithms. Fake websites can also be detected automatically with a low false positive ratio by comparing them with the real websites. Clients can utilize anti-phishing-capable devices or software such as anti-virus, anti-spam, anti-spyware, or IDS. Web browsers can be armed with anti-phishing plug-ins such as Spoofstick or SpoofGuard. Given the damage that can potentially done by a phishing attack, a diverse range of efforts are being made to protect ordinary users (such as in user education, reporting and response and legal protection).

The outlook is not entirely bleak against phishing given the technical and social remedies being pursued. If organizations prepare well, remain vigilant and follow attack trends carefully, they can respond quickly and effectively with a range of techniques to defend their customers' data. If individuals take a responsibility for their protection and adopt a defense-in-depth approach, they can shield themselves against the most sophisticated attacks. Although there is no simple solution, active and aware users and organizations have the ability to form a strangle-hold on this ever-growing threat.

1. Introduction

Phishing attacks attempt to fraudulently solicit sensitive information from a user by masquerading as a known trustworthy agent [5,60]. They most commonly use 'spoofed' emails in association with fake websites in order to coerce a user into revealing personal financial data, such as credit card numbers, account user names and passwords, or social security numbers [49]. By masquerading as well-known banks, e-retailers and credit card companies, phishers often convince recipients to respond [5]. Phishing attacks range in sophistication, from simply fooling a user with a seemingly legitimate communication, to deliberately exploiting weaknesses in software to prevent users from determining the true nature of the attack.

The idea of obtaining user information through fraudulent means is not unique; phishing is merely a subset of two larger problems that exist in both the electronic and 'real-world' domain:

- Social engineering: is any attempt to obtain confidential information by manipulating legitimate users. Phishing uses email and counterfeit websites to achieve this goal [61]. While most Internet security threats take advantage of software vulnerabilities, this attack exploits trust relationships previously developed[1] between the user and other users or organizations.

[1] In some circumstances, the trust relationship is created and then immediately abused. For example, an attacker might attempt to reset login credentials from an organization's helpdesk by convincing the tech-

- Identity theft: uses the information gained through techniques such as social engineering in a deliberate attempt to use another person's identity. This can then be used, for example, to gain access to their finances or frame them for a crime [59]. Techniques used involved include stealing mail, rummaging through garbage ('dumpster diving'), stealing personal information from computer databases, or infiltrating large organizations that store large amounts of information. Phishing is merely a mechanism of obtaining this information.

Phishing shares many characteristics with two similar techniques: pharming and the abuse of alternate data streams. Both require a higher level of skill to execute successfully than simple phishing schemes. Pharming is a more active form of phishing, with the user automatically directed away from the legitimate website to the fraudulent website without warning [9]. Alternate data streams can be used to secretly associate hostile executables with legitimate files; this is effectively 'file phishing' [8]. With minimal effort, a hidden executable can be masked and its function obscured. Like phishing, the resulting environment is not entirely as it appears.

Another related technique is an independent scam website that lures victims through voluntary web navigation or through a search engine instead of using active emailing. An unsuspecting user may buy a product from a scam website, or make an investment on a foreign company through a scam website. While its effect is similar to phishing, the process of luring the victims is different. A scam website is a passive form of phishing, silently waiting for a prey, but it can become more effective when supplemented with phishing techniques. Many anti-phishing techniques covered in this chapter are also useful for identifying those independent scam websites, for example, Trustbar (see Section 4.4) displays the logos and certificate authority of the website.

1.1 History

The word 'phishing' is a derivative of the word 'fishing' and describes the process of using lures to 'fish' for (i.e., obtain) sensitive user information [2]. Exchanging 'f' for 'ph' is a common hacker replacement; it is most likely an acknowledgement of the original term for hacking, known as 'phreaking'. The original form of hacking, known as phone phreaking, involved sending specific tones along a phone line that allowed users to manipulate phone switches. This allowed free long distance calls, or the billing of services to other accounts, etc.

The first recorded use of the term 'phishing' was in January 1996, in a posting to the alt.2600 newsgroup by drspamcake@aol.com. It was in reference to the theft

nician that they are a legitimate user in some kind of unusual situation that requires standard procedures to be bypassed.

of AOL user accounts [13] by scamming passwords off unsuspecting users. The technique itself predates the reference by drspamcake@aol.com: AOL users were already being targeted via instant messages sent by users masquerading as AOL staff members, who would request a user's account details [60]. By 1995, AOL software contained a 'report password solicitation' button, which gives an indication of the magnitude of the threat.

Those seeking free AOL accounts initially took advantage of poor credit card validation techniques and used algorithmically generated credit card numbers to acquire accounts that could last up to a month. They turned to phishing for legitimate accounts after AOL bought in measures in 1995 to prevent this type of behavior. Hacked accounts were referred to as 'phish' and by 1997, phish were being actively traded as a form of electronic currency [46]. For example, phish could be traded for hacker software or 'warez'.

Since that time, the definition of phishing has widened to cover not only obtaining user account details, but also obtaining access to all personal and financial data. The sophistication of the field has also grown: modern schemes go far beyond simple instant messages, and typically target thousands of users using mass mailings and fake websites.

1.2 Current Status

Phishing is now more than a mere annoyance: it is a common online crime that is relatively easy to perform, has a low chance of being caught, and has a potentially very high reward [27]. It is for these reasons that phishing has been embraced by organized crime, both in the United States and in Eastern Europe (particularly in Russia and the former Soviet bloc). It is also believed [23] that terrorist sympathizers, operating out of Africa and the Middle East, are using phishing to steal identities and cash.

Phishers typically send out massive emails in the hope that some naïve recipients will respond. Although the majority of the recipients feel suspicious on such phishing emails, some recipients are successfully convinced into the scam. Phishers successfully attacked an estimated 1.2 million users and cost an estimated US$929 million in the twelve months to May 2005 [38]. US businesses lose an estimated $2 billion a year as their clients become victims [39]. The Anti-Phishing Working Group[2] [3]

[2] Anti-Phishing Working Group is a global association of industrial and law enforcement organizations focused on eliminating the fraud and identity theft that result from phishing, pharming, and email spoofing. It includes more than 1,600 companies and agencies worldwide including 8 of the top 10 US banks and 4 of the top 5 US ISPs. It offers anti-phishing education, maintains phishing data, evaluates the anti-phishing methods, and work with law enforcement and legislature. Its website is available at http://www.antiphishing.org.

received 20,109 reports of phishing scams in May 2006, primarily targeting financial institutions (92% of all reports). In year to May 2006, the average growth rate for phishing attacks was 34% [3].

1.3 Phishing Illustrated

There are several steps that phishers follow. Two examples are illustrated here, for posers and mongers, respectively. The posers are the bottom-feeders in the phishing community that exhibit a very low level of sophistication. The phish mongers are those who deploy these phish scams in such a way that they stand a measurable chance of success against a reasonably intelligent and enlightened end-user [7].

1.3.1 Posers

The essential requirements of effective phishing require that the bait:

1. look real;
2. present itself to an appropriate target-of-opportunity;
3. satisfy the reasonableness condition (i.e., going after the bait is not an unreasonable thing to do);
4. cause the unwary to suspend any disbelief;
5. clean up after the catch.

Figure 1 is modeled after some live phish captured on the net and meets all of the five criteria identified above. First, the email looks real—at least to the extent that it betrays nothing suspicious to a typical bank customer (a.k.a. target-of-opportunity). The graphic appears to be a reasonable facsimile of a familiar logo, and the salutation and letter is what we might expect in this context. Second, the target is the subset of recipients who are Bank of America customers. The fact that the majority of recipients are not is not a deterrent because there is no penalty for over-phishing in the Internet waters. Third, the request seems entirely reasonable and appropriate given the justification. Customers reason that if they were a bank, they might do the same thing. Fourth, the URL-link seems to be appropriate to the brand. Unwary Internet users might readily trade off any lingering disbelief for the opportunity to correct what might be a simple error that could adversely affect use of a checking or credit card account. The link to "verify.bofa.com" may be assumed to take us to an equally plausible web form that would request an account name and password or PIN.

The unwary in this case is M. Jones whose harvested web form appears to the phisherman as in Fig. 2. This is a screenshot of an actual phishing server in our lab.

In order to complete the scam the fifth condition must apply. In this case, after the private information is harvested, the circle is completed when the phishing server

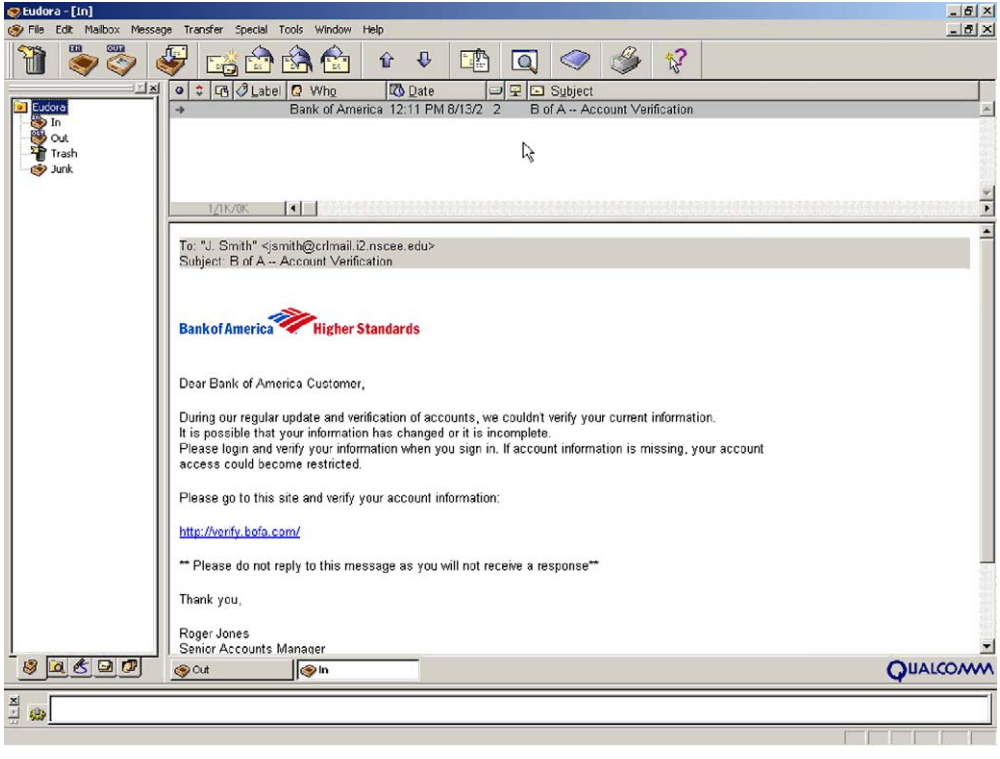


FIG. 1. Phishing email that satisfies our five effectiveness criteria.[3]

[3] Any legitimate emails from Bank of America (as well as most other corporate names used in this chapter) to a customer never put a reply URL in the message. Instead they ask the customers to go to their corporate website directly, avoiding a direct response to what may be a fraudulent email.

FIG. 2. Phishing from phisherman's perspective.

redirects the victim to the actual bank site. This has the effect of keeping the bank's server logs roughly in line in case someone makes an inquiry of the help desk. Figure 3 illustrates this activity.

1.3.2 Mongers

Mongers employ more sophisticated schemes. Look carefully at the cursor in Fig. 4. The cursor seems to be sensing the link even though it is not particularly close to it. The fact is that it is not sensing the link at all, but rather an image map.

A quick review of the source code, below, leads us to a veritable cornucopia of trickery.

```
<x-html>
<html><p><font face="Arial"><A
HREF="https://signin.ebay.com/ws/eBayISAPI.dll?SignIn&sid=verify&co_p
artnerId=2&siteid=0"><map name="xlhjiwb"><area coords="0, 0, 646,
569" shape="rect" href="http://218.1.XXX.YYY/.../e3b/"></map>
<img SRC="cid:part1.04050500.04030901@support_id_314202457@ebay.com"
border="0" usemap="#xlhjiwb"></A></a></font></p>
<p><font color="#FFFFF3">Barbie Harley Davidson in 1803 in 1951 AVI
</x-html>
```

Several features make it interesting. First, the image map coordinates take up nearly the whole page. Second, the image that is mapped is the actual text of the email. So what appeared to be email was just a picture of email. Thus, the redirect was actually not a secure connection to eBay at all as it appeared, but an insecure connection to `218.1.XXX.YYY/.../e3b/`. While Windows users see the "dots of laziness" frequently when path expression is too long for the path pane in some window, this is not a Windows path in a path pane. These "dots of laziness" are a directory name. It is not clear why someone would create a directory named "..." as

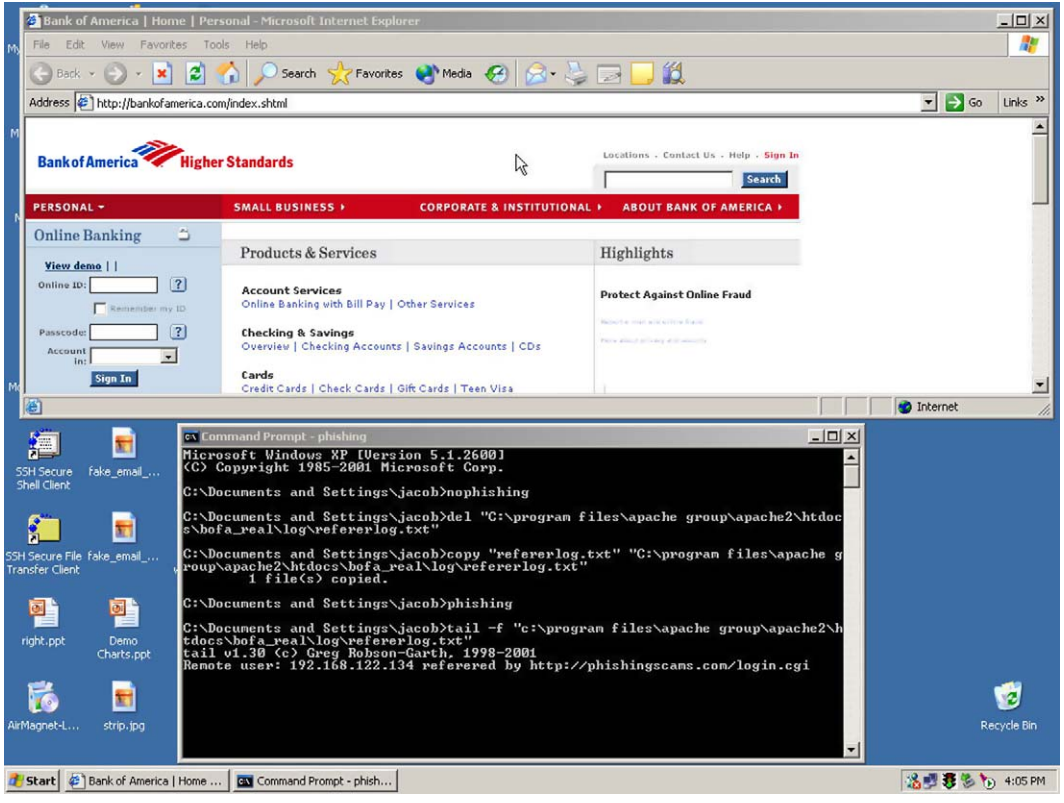
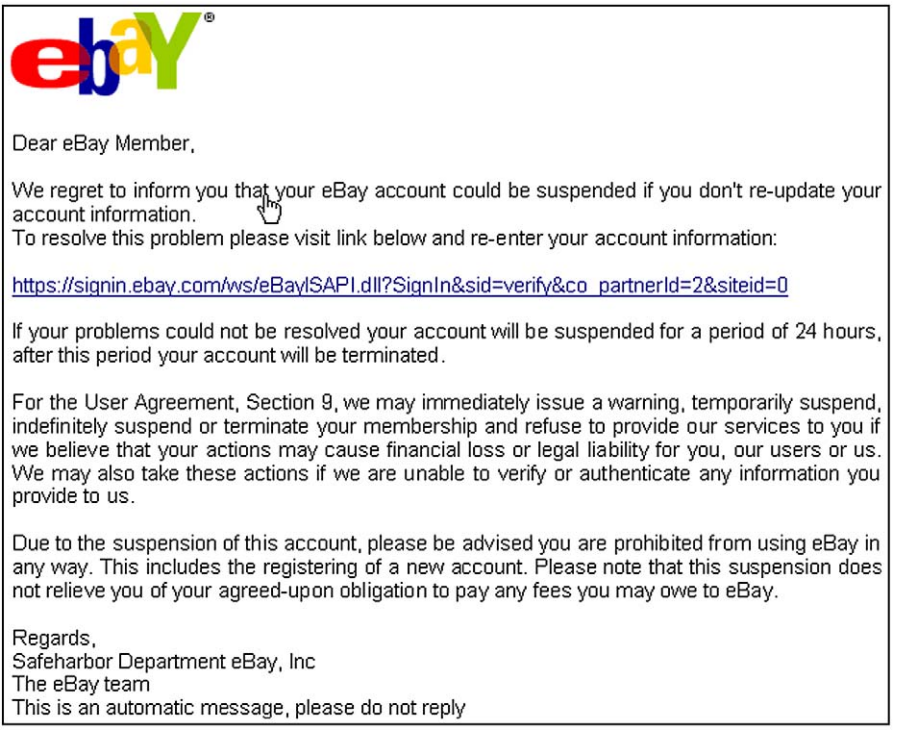


FIG. 3. Phish clean-up.



Dear eBay Member,

We regret to inform you that your eBay account could be suspended if you don't re-update your account information.
To resolve this problem please visit link below and re-enter your account information:

https://signin.ebay.com/ws/eBayISAPI.dll?SignIn&sid=verify&co_partnerId=2&siteid=0

If your problems could not be resolved your account will be suspended for a period of 24 hours, after this period your account will be terminated.

For the User Agreement, Section 9, we may immediately issue a warning, temporarily suspend, indefinitely suspend or terminate your membership and refuse to provide our services to you if we believe that your actions may cause financial loss or legal liability for you, our users or us. We may also take these actions if we are unable to verify or authenticate any information you provide to us.

Due to the suspension of this account, please be advised you are prohibited from using eBay in any way. This includes the registering of a new account. Please note that this suspension does not relieve you of your agreed-upon obligation to pay any fees you may owe to eBay.

Regards,
Safeharbor Department eBay, Inc
The eBay team
This is an automatic message, please do not reply

FIG. 4. Phish mongering.

it certainly falls short of the mnemonic requirements most of us learned in intro to programming.

On the other hand, it might blend in stealthily with the other *nix hidden files, "." and "..", and possibly escape an onlooker's suspicion. This suggests that the computer at the end of `218.1.XXX.YYY` may not be the phisher at all, but another unsuspecting victim whose computer has been compromised (for that reason, the final two octets of the IP address have been concealed). Another sign of intrigue is the font color of almost pure white "`#FFFFF3`" for "Barbie Harley Davidson in 1803 in 1951 AVI." Though their names are sullied, neither Barbie nor Harley Davidson had anything to do with this scam. This white-on-white hidden text is there to throw off the Bayesian analyzers in spam filters. As the email text is actually a graphic, the Bayesian analysis likely concludes that this is about Barbie and her Harley given that it has no other text to base its decision on. As opposed to the posers, this phish monger is moderately clever.

So far the most common of phishing attacks have been illustrated. The rest of this chapter is organized as follows. In Section 2, the fundamental techniques used in phishing are explained, such as bulk emailing, alternative delivery techniques, and obfuscation techniques for masking the fake websites. Section 3 addresses advanced phishing techniques, including Malware, man-in-the-middle attack, and website-based attacks, etc. In Section 4, anti-phishing techniques are discussed; however, technical solutions are only part of the picture in anti-phishing efforts and in Section 5, more comprehensive efforts are examined. The chapter is then concluded in Section 6.

2. Core Phishing Techniques

In order to achieve their goals, phishers typically use a mixture of two techniques: social engineering and technical subterfuge [5]. Social engineering is the primary technique used and appears to some extent in most attacks. Arguably, the use of this technique distinguishes phishing from other forms of electronic fraud. Technical subterfuge exploits software-based weaknesses in both servers and clients in order to mask the true nature of the transaction from the victim or plants crimeware onto PCs to steal credentials directly (often using Trojan keylogger spyware). Pharming crimeware misdirects users to fraudulent sites or proxy servers, typically through DNS hijacking or poisoning [5].

Both techniques are employed in the pursuit of the same goal: the victim must be convinced to perform a series of steps to reveal confidential data. However, these two techniques seek to attack from opposite ends of the spectrum: one targets human weaknesses, the other technical vulnerabilities. In this section and the next, how phishers exploit these vulnerabilities is examined.

2.1 Bulk Emailing Combined with Fake Websites

A basic phishing scheme needs four elements: a bulk mailing tool, a standard email, a ghost (fake) website and a database of email addresses [20,49]. Typically, the ghost website is set up, and then the bulk email tool distributes the phishing email to all those addresses in the email database. The most successful phishing scams have genuine looking content in both their e-mail (if used) and the fake website. This includes:

- using images from the real website;
- in the e-mail, use official text (social engineering techniques apply here);
- many phishing sites simply copy the real website (using `wget` or the like).

The standard email sent is branded so it appears as though it was sent by a trusted and reputable party (e.g., a financial institution). The most commonly spoofed companies include Citibank, eBay and PayPal. It is likely that not all those users within the email database will have accounts with the spoofed organization; this is somewhat unavoidable and it can reveal the operation of a phishing attack. A number of techniques can be used within the email [20,34,46,49]:

- The email will use authentic logos and graphics obtained from the legitimate website in order to imitate the company's visible branding (see Fig. 5). The email itself is likely to be a modified copy of a previous corporate mailing.
- It will also use a spoofed 'mail from' address to make the email appear to originate from the proper domain. This is a well-known flaw in the SMTP protocol: phishers can set the 'mail from' and 'reply to' headers to an email address of their choice.
- It will typically contain a URL that appears to link to the legitimate site; however, the URL will likely relay the user to the ghost website (the URL in Fig. 5 sends the user to `http://218.246.224.203/`). This obfuscation will generally require the use of HTML email. The use of HTML formatting also allows the attacker to create a more authentic email by using legitimate graphics. This would allow the attacker to include a HTML form inside the email itself to solicit user information, although this is relatively uncommon.
- Using an HTML email to imitate a plain-text email can further increase the difficultly an average user faces in identifying the hidden qualities of the email (see Fig. 5).
- The email is also likely to contain URLs that refer the user to the legitimate site (for example, to a help or contact page) in order to better mimic a mailing from the legitimate organization (see Fig. 5).
- Assurances are included within the email to gain trust, such as "we will not ask you for sensitive personal information. . . in an email" or the use of the **TRUSTe** symbol (which identifies organizations with a high level of personal information protection). Other security assurances are also used. For example, that the email is free of viruses and is not spam.
- The HTML code behind the email is usually long, which limits the ordinary user's ability to locate and check the actual target of the URL contained in the email (if they decide to do so).

The key objective of the email is to create a plausible premise that persuades the user to release personal information. The contents of the email must be designed to illicit an immediate user reaction, which prompts them to follow the enclosed link to the website (see Fig. 5). For example, the email may [20]:

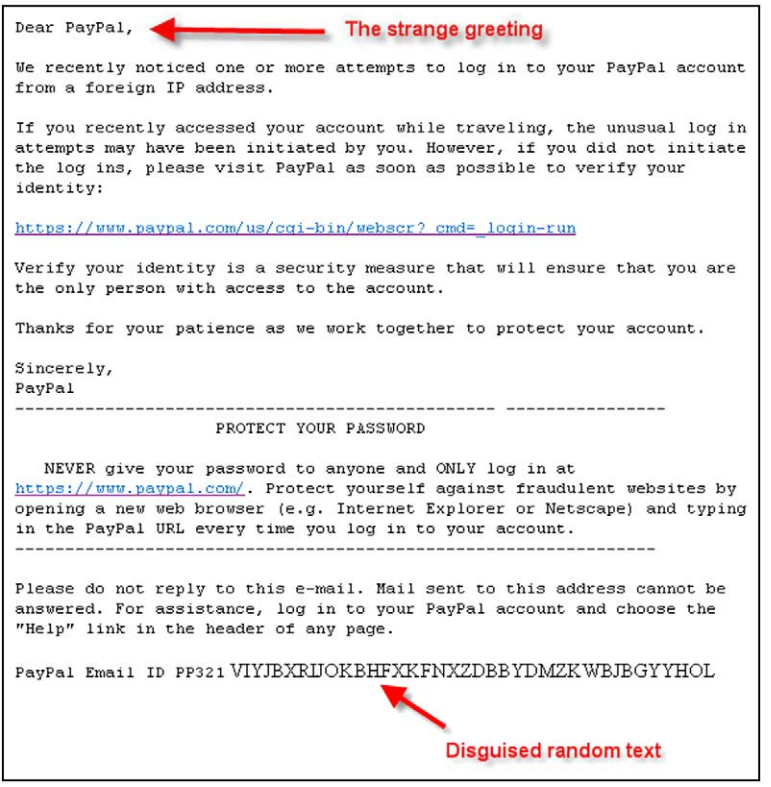


FIG. 5. An example phishing email recorded by the Anti-Phishing Working Group (10/05/2005).

- refer to an unauthorized transaction from the victim's bank account,
- reveal the user has won a prize,
- indicate the organization has lost their account details, requiring the user to manually update them, or
- threaten to charge a fee without an immediate reply.

Ironically, many phishing emails take advantage of the user's fear of online fraud [20], using a premise that requires users to update their information due to a security system upgrade or similar (see Fig. 5). While there are many different approaches, each must create a scenario to convince the user to provide the requested information in a timely manner (i.e., before the phishing site is shutdown).

Using a bulk email tool and an email database, the standard email can be sent to millions of legitimate, active email addresses within a few hours. The use of a network of trojaned machines can speed the process up considerably [34,46]. The email database can be acquired from a number of sources on the Internet, for free or for a fee. Such vendors target their email databases at spam distributors; however, the databases they distribute are equally useful for both purposes.

A ghost, or fake, website is typically hosted by a hijacked PC,[4] compromised by other means [34,46,49]. A mechanism will need to be in place to facilitate information retrieval (by the phisher); it is speculated anonymous login or email may be used for this activity. Ideally, this machine would reside in a different country to that of legitimate website's organization as this increases the difficulties involved in shutting the phishing website down. The domain name and email URL are designed to prevent the target from noticing they are transacting with a ghost website rather than the legitimate site. Subtle character replacements can achieve this: for example, `www.paypal.com` could be imitated using `www.paypa1.com` (note the "one" in the name) or `www.paypal.cc`. More sophisticated methods will be discussed in Section 2.3 (URL obfuscation).

The content of the website is likely to be a near-exact copy of the legitimate site, updated to allow the phisher to record user details. The ghost website is also likely to contain introduction pages, processing pages and pages thanking the user for submitting their data, in a further attempt to increase authenticity. It may also use a legitimate server-side certificate, signed by Verisign or similar, issued to the ghost website. Alternatively, it could use an unsigned certificate under the assumption that most users will be unable to interpret the security warning (if the security warning has not already been disabled). Even invalid or fake certificates are likely to make users feel more secure. The absence of SSL/TLS security may alert some users to the true nature of the website; however, security indicators within the user's browser can potentially be forged using browser exploits (see Fig. 6).

Upon submitting their details to the ghost website, the user is often redirected to the legitimate site to encourage the user to continue to believe they have revealed their personal data to a legitimate organization. Alternatively, the phisher may use a post-submission page to encourage the user not to access or use their accounts for a specific timeframe, in order to mask the phisher's exploitation of their sensitive information (e.g., use of a credit card number). It is critical that the user does not realize they have submitted their data to an illegitimate organization. If this occurs, the personal data can be quickly rendered useless (e.g., their password will be changed or accounts closed).

[4] This can sometimes be detected by the use of a non-standard HTTP port embedded in the target URL.

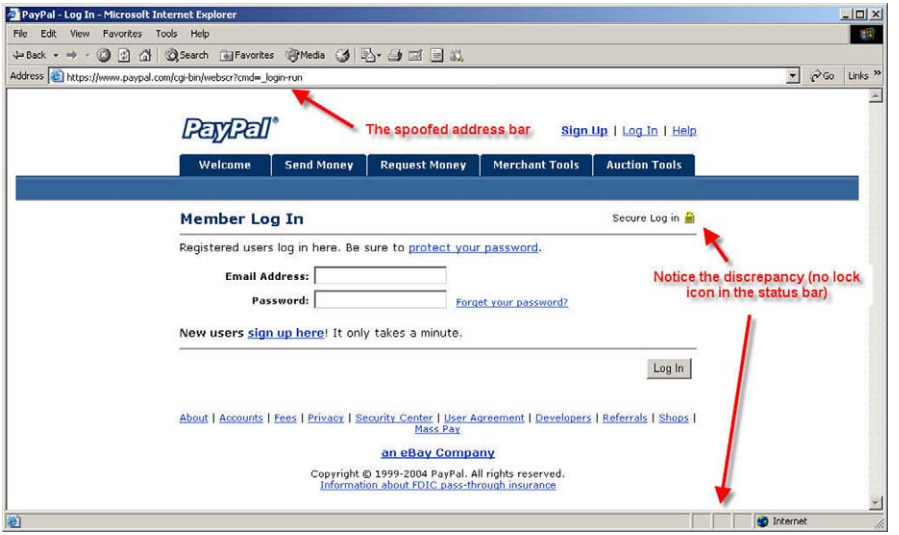


FIG. 6. A phishing website targeted at PayPal customers from the Anti-Phishing Working Group.

2.2 Alternative Delivery Techniques

Communication via email remains the most common and successful form of attack; however, other electronic communication mechanisms are becoming increasingly popular, such as web pages, IRC and instant messaging [46]. In all forms of communication, the phisher must imitate a trusted source in order for the victim to release their information.

2.2.1 Web-based Delivery

Rather than distributing the malicious URL (or similar) via email, an increasingly popular technique is to place it in website content [46]. The website itself can be hosted by the phisher, or by a third party host (which could be acquired freely, for a fee, or via a Trojan horse attack). The level of sophistication varies: a malicious URL could be disguised and placed on a popular website or comment board, or a website developed for the express purpose of luring in potential victims.

If specialist website is employed to lure victims, the phisher may employ several techniques [46]:

- Hidden items within the page (e.g., tiny graphics) may be used to identify suitable victims.
- Pop-up or frameless website may be used to hide the true source of the website.
- Malicious content may be embedded to exploit a vulnerability of the user's browser. This exploit can be leveraged to install software onto the user's computer without their knowledge. For example, software such as key loggers, screen grabbers, back doors or other trojan horse programs may be installed.
- Trust relationships inside the user's browser configuration may be abused in order to allow scriptable components or access data storage areas.

In order to attract users to their website, fake banner advertising could be employed. Banner images belonging to the company the phisher is attempting to mimic can be placed on popular websites and direct users to the phisher's website, rather than the legitimate website. Standard URL obfuscation techniques can be used to hide this subtle redirection from the user. Many vendors provide online registration for banner advertising; with a stolen credit card (or similar), a phisher can easily acquire advertising while remaining concealed from law enforcement agencies.

2.2.2 IRC and Instant Messaging (IM)

While these techniques were popular in the early days of phishing, email has become the technique of choice for modern day phishers. However, it is predicted [27, 46] that the use of these techniques will become more popular in the future, given that these technologies are popular with home users and are gaining in their complexity on a regular basis. Embedded dynamic content, such as multimedia, graphics, and URLs, can now be sent with many IM programs, allowing standard email and web-based phishing techniques to be easily mapped to this domain. Automated bots, that interact unsupervised with IRC participants, could also be used by phishers to coerce users into visiting their fake websites.

2.3 Obfuscation Techniques

In addition to the techniques previously mentioned, phishers have other techniques to deliberately disguise the true nature of the message from the recipient.

URL obfuscation is an essential part of most phishing attacks. It fools the user into believing they are following a link to a legitimate website; in actual fact, they are being transported to the phisher's fake website. The simplest technique for URL obfuscation uses HTML; the legitimate website's address is displayed to the user in plain text, but the link is targeted at the phisher's website. For example:

```
<a href=''http://www.evilsite.com''>http://www.citibank.com</a>
```

is displayed to the user as:

http://www.citibank.com

but links to `http://www.evilsite.com` (see Fig. 5). This technique would fool a basic user, who many not be aware than the display address of the hyperlink can be different to its target. Other URL obfuscation techniques include [20,46,49]:

- Simple character replacement can obfuscate URLs, as using the legitimate URL as a prefix to another domain (e.g., adding bank.com to `www.citibank.com` to form `www.citibank.com.bank.com`). Variations of the legitimate domain name can also be used (e.g., `www.citibank-accounts.com`). All of these simple techniques would go unnoticed by an inexperienced user.
- An extension of the aforementioned technique involves the vulnerabilities of the ASCII character set. Foreign characters are encoded using 2-byte Unicode rather than 1-byte ASCII. Attackers can utilize visually similar characters in different Unicode sets to exploit confusion, which is called Unicode attack [26]. Domain names can be registered in different languages: some foreign character sets look identical to ASCII characters, but are interpreted differently during the domain name lookup process. According to Fu et al. [26], there are eight possible representations of alphabet character "s", "o", "u", "p" and so on. Users may not be able to distinguish the differences at a quick glance. A recent scam allowed `microsoft.com` to be registered, using the Cyrillic 'o' instead of the ASCII version; visually these characters are identical.
- Most browsers also accept alternative encoding schemes for hostnames, in order to allow support for local languages.
  - Escape encoding allows the inclusion of characters that may need special syntax in order to be correctly interpreted (e.g., a space in a URL string may indicate the end of the URL or it may be part of the URL). These are included as %xx, where xx is the hexadecimal ASCII code for the character. This also allows normal characters to be encoded in this way (e.g., %41 is 'A' and %20 is a space).
  - Unicode encoding allows characters to be stored in multiple bytes. This permits a far greater number of characters (65,536) that can be encoded in comparison with ASCII (128), and allows a unique identifier for every character no matter what language or platform. In a Microsoft Windows environment, these characters can be encoded as %u0000, where 0000 is the hexadecimal code for the character (e.g., %u0056 is 'V').
  - UTF-8 encoding is a commonly used format of Unicode, and preserves the full ASCII character code range. This allows standard characters to be en-

coded (and obfuscated) in longer escape-coded sequences (for example, '.' can be encoded as %F8%80%80%80%AE).

- Multiple encoding can occur when applications incorrectly parse escape-encoded data multiple times and at multiple layers of the application. This vulnerability can be exploited by phishers encoding characters multiple times and in different fashions (e.g., %%35%63: the second part of the string, '%35%63', decodes to '5C'. This string, combined with the prefix '%', gives '%5C', which decodes to '\').

- The standardized URL encoding format allows for the insertion of a username and password within the string (e.g., `http://username:password@mysite.com`). Effectively, everything between the protocol name and the '@' character is ignored; this allows the construction of obfuscated URLs such as `http://citibank.com:mybank@fakesite.cc`. Due to the threat this encoding format presented, some browsers no longer allow links of this form (such as Microsoft Internet Explorer).
- Some online websites provide redirection URLs: these allow the construction of URLs that give no indication of the actual target. Redirection URLs forward users onto another predefined site when they are accessed. For example, the link `http://r.aol.com/cgi/redir?http://jne9rrfj4.CjB.neT/?uudzQYRgY1GNEn` was found in a Citibank phishing attack, and includes a double redirect. The browser is first sent to `r.aol.com`, and then redirected to `jne9rrfj4.cjb.net`, which redirects the user to the fake website.
- The website host name can be obfuscated by encoding it as an IP address rather than a domain name. The use of a standard decimal IP address in place of the host name will go provide some measure of obfuscation; however, encoding the IP address in dword (e.g., `http://3532038435`), octal (e.g., `http://0322.0206.0241.0043`), hexadecimal (e.g., `http://0xD2.0x86.0xA1.0x23`) or a mixed format (e.g., `http://0322.0x86.161.0043`) will confuse even more users.

Given the bulk nature of these emails, and the threat they pose to users, most organizations opt to treat them as spam, and filter them before they reach users. Several techniques can be used by phishers to make the filtering task more difficult, and therefore reach more potential victims [46]:

- Text can be obfuscated to avoid spam filter detection. For example, lowercase 'L's could be replaced with upper-case 'I's, both of which appear visually similar to humans, but are interpreted quite differently by software. The similarity between the letter "`l`" and the numeric "`1`" may also be exploited as in `www.ao1.com` instead of `www.aol.com`.

- Where possible, the phisher may seek to personalize the email to the intended user, or at least make it unique (e.g., by inserting random text. See Fig. 5 for an example of this). This will largely depend on the email database used. By ensuring each email is unique will make it more difficult for the email to be filtered by hash-based anti-spam techniques (e.g., Cloudmark [14]).
- The use of HTML email allows the spammer to hide random words within the email (see Fig. 7). They can be included as comments or colored white to avoid detection by the user. These hidden words can make the email seem more legitimate to the spam filter, without altering the message eventually viewed by the user.

JavaScript can be used by the phisher to execute a number of attacks. In terms of obfuscation, it can be used to further hide the true destination of the link from the

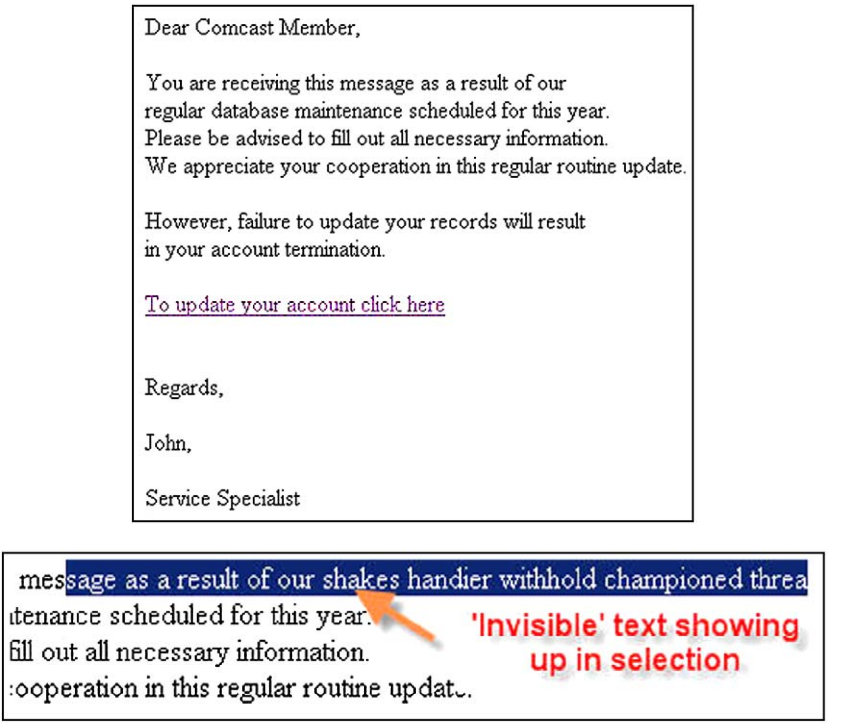


FIG. 7. An example from the Anti-Phishing Working Group that illustrates the use of hidden text in order to avoid detection by spam filters.

user. Most email and browser applications show the true destination of the link in the status bar at the bottom of the windows when a user moves their mouse over the link; this behavior can be overridden, so the legitimate link is displayed rather than the link to the fake site.

Many of the obfuscation techniques and the attack techniques discussed here can be readily identified if the user is sophisticated enough to read and interpret the source code. JavaScript can be used to deter the user from doing so as it can disable the page's right click menu. Those users who right click can instead by greeted with a pop-up message box (with a copyright notice or similar). However, this does not prevent the user access the page source from alternative locations (such as through the browser's menu bar, if it is present).

3. Advanced Phishing Techniques

3.1 Malware

Malware is a term used to describe any form of malicious software, including viruses, worms, trojans and others. For phishers, this software represents a new route to defraud their victims that may complement or even replace the social engineering techniques that phishing often relies upon. The potential for fraud here is greater: rather than asking the victim for their information, they simply take it. The complete replacement of social engineering in a phishing attack with malware arguably represents an entirely different class of attack. However, much malware still relies on the targeted user to approve its installation and/or execution; it is for this reason social engineering is likely to remain a core skill relied upon by phishers.

Known malware worms used for phishing include [42]:

- W32.Mimail.I,J,P,Q,S: these worms attempt to fool users into revealing credit card information in response to a Microsoft Windows expiration notification or a PayPal application. The requests are displayed as web pages served from the local machine. Typically, the worm is attached to an email message (passive worm), and a social engineering approach is used to encourage the user to run the attachment. The worm is then copied to the local drive, where it embeds itself in the machine's startup routine. It is self-propagating: it searches the user's documents for email addresses, and sends itself to all found addresses. Some variants also attempt to retrieve other information (Internet account information, RAS phone book entries, E-Gold information, and other personal information such as the user's credit card numbers, their birthday and their social security number) and relay it to the phisher using HTTP POST/GET.

- Backdoor.Lala and Backdoor.Lala.B: are trojan horses that permit unauthorized access to remote computers. They also attempt to steal confidential information (such as cached passwords and cookies), log keystrokes, and allow remote file execution. Cookies associated with financial institutions, such as PayPal, E-Bullion, Evocash, WebMoney and various banks, are targeted.
- PWSteal.Bancos and W32.Bibrog: are worms designed to monitor websites visited by the user. If the user attempts to visit particular bank websites in their browser, the worm redirects them to a phishing site. This website records and steals the user's information. It replicates by sending itself to all email addresses in the user's Outlook contacts folder.
- More recently, the Korgo [56] worm was used to infect unpatched systems with a keylogging trojan designed to steal online banking information and securely relay it back to its creators. It collected any data entered into a web form by the user. Mikko Hypponen, of F-Secure, advised users infected by Korgo to change all their passwords and to cancel their credit cards. "This is not a joke," he said.
- Other trojans and worms are known to record keystrokes or record data entered in web form input fields (e.g., W32.Dumuru.Y, W32.Dumuru.Z, PWSteal.Tarno, PWSteal.Banpaes, PWSteal.Banpaes.B, the TROJ_WINCAP series and W32.Mimail.C).

Interestingly, Brazilian banks appear to be over-represented in malware-based phishing schemes. In the six months to March 2004, twenty different malware applications were identified that targeted Brazilian banks [42].

Malicious users have long used software designed to log keystrokes and record screen captures to obtain sensitive data. These utilities are being employed more frequently in phishing attacks. These utilities can remain on a user's computer for an indefinite amount of time, and can record a far greater amount of information than any one basic phishing attack. Depending on the extent to which the attacker is willing to analyze the recorded logs, account information from a variety of sites can be harvested (rather than a single account typically recorded by a standard phishing scheme). Given the volume of data these techniques can potentially generate, the attacker has three options to retrieve recorded information:

- Data streaming: data is sent to the attacker as soon as it is generated. This requires a continuous connection between the attacker and victim.
- Batch collection: information is uploaded to the attacker's server on a regular basis, using FTP, HTTP, SMTP or similar.
- Backdoor collection: remote access software is placed on the victim's computer to allow the attacker to connect and download the recordings on demand.

Key loggers record all keystrokes entered by a user. With the use of appropriate filtering techniques, the attacker can isolate credentials used to access various online services. They vary in sophistication: some will record all key presses, while other will only record key presses entered in the web browser. The Anti-Phishing Working Group [3] recorded 215 phishing attacks in May 2006 (out of a total of 20,109) that used key logging malware.

Screen capture utilities were, in part, a response to advanced anti-key logging techniques used by some organizations; they record the other primary form of user input. These utilities record a screen image on a regular basis, or part of a screen image (i.e., the relevant observational area, such as the authentication area of a particular website). Partial screen captures minimize the size of the require upload to the attacker. Such techniques are successful against organizations such as Barclays Bank; they require users to select several, randomly selected, characters from their 'memorable word' from drop-down lists (e.g., the second and fourth letter) as part of their technique.

Phishers and spammers typically share some commonalities: both typically want to distribute substantial amounts of email quickly and without being detected. There is some evidence [49] that these groups are exchanging techniques. Some techniques applicable to phishing are [42]:

- Spam relays: are machines that accept email and forward it on to another SMTP server. The trojan horse Backdoor.Hogle turns unsuspecting machines into spam relays. By 'recruiting' a number of spam relay machines, the phisher could send messages quicker and make it more difficult for authorities to trace the message's origin. It is estimated that phishers can use up to 1,000 computers in their attacks [29].
- Reverse HTTP proxies: are used to hide the true location of the web server. The domain name included in the mass email message is configured to point to the IP address of a machine running the reverse HTTP proxy (such as a machine infected with the Backdoor.Migmaf trojan horse). The machine proxies any HTTP requests back to the actual web server, and sends any responses back to the client; at no time does the client know the IP address of the actual web server. Additionally, the IP address that the hostname points to is changed on a regular basis, making locating and neutralizing the actual web server particularly difficult. The Backdoor.Migmaf trojan has been used in a PayPal phishing scam.

3.2 Man-in-the-middle Attacks

The principles behind man-in-the-middle attacks are simple: the attacker acts as an intermediary between the victim and the legitimate site and records the information

exchanged between the two parties [46]. The attacker achieves an ideal vantage point on the transaction, and can potentially remain unnoticed by both parties. The idea behind this attack is not unique to this domain: it is used throughout network security (e.g., TCP hijacking).

The intermediary machine utilized by the attacker is referred to as the proxy. Ideally, it is transparent: it does not effect the communication between the legitimate parties, and is not easily detected. Such proxies can be located on the same network segment as the target, or en route to the legitimate website. To ensure the client routes traffic through the proxy, browser proxy settings can be overridden (either by a software exploit, or through the use of social engineering); however, this is now obvious to the client. Proxy configuration is generally performed before the phishing email message is sent: this ensures the transmitted data is recorded if the user follows the enclosed link.

This form of attack can be successful for both HTTP and HTTPS (i.e., SSL/TLS) connections [46]. SSL/TLS provides application-level security between the client and the legitimate website; standard proxies between these two parties can only record the cipher text. However, if the phishing email can ensure the user connects to the proxy, rather than legitimate website, their data can be recorded. URL obfuscation techniques are useful in achieving this. The proxy passes all of the user's requests to the legitimate website, and responses from the legitimate website are passed back to the user. In the case of a SSL/TLS connection, a secure connection is established between the proxy and the legitimate website. A secure connection can also be maintained between user and the proxy via the methods described previously.

Using the legitimate website to process information submitted by the victim also aids the phisher as it allows invalid data to be discarded. It not only makes the phisher's job of identifying valid accounts easier but it also makes the site appear more authentic to the user [20].

DNS cache poisoning [46] attempts to corrupt the local cache maintained by a specific DNS server. When a user requests the IP address of a domain name, the request is forwarded to the DNS server. If the DNS server does not have the IP address of the domain in its cache, it will query an authoritative domain name server for the information. The BIND attack, an example of DNS cache poisoning, requires the attacker to spoof the reply from the authoritative name server; in the reply, the attacker can set the IP address of the queried domain to any desired machine. By exploiting DNS vulnerabilities, the phisher could potentially redirect traffic directed at a site such as www.citibank.com to their fake website. DNS cache poisoning can be particularly effective, as most ISPs operate one DNS server for all of their subscribers. If the network's DNS server is poisoned, all of the ISP's customers will be redirected to the fake website.

3.3 Website-based Exploitation

After the user has been successfully lured to the fake website, the phisher has a variety of technologies to further disguise and obfuscate the true identity and nature of the website. Website scripting and markup languages such as HTML, JavaScript, DHTML, ActiveX, VBScript, etc. give the phisher tremendous power to completely mimic the appearance of the legitimate website [46].

HTML frames can be used to obscure attack content. They enjoy wide browser support and are simple to use, and therefore are ideal for phishing websites. For example:

```
<frameset rows="100%,*", framespacing="0">
  <frame name="real" src="http://www.citibank.com" scrolling="auto">
  <frame name="hidden" src="http://fakesite.com" scrolling="auto">
</frameset>
```

The legitimate Citibank site is all that is viewable within the browser window; however, this code snippet also loads HTML from fakesite.com. The additional code could [46]:

- deliver additional material, such as overriding page content or graphics,
- retrieve session IDs,
- execute screen captures, log keystrokes or monitor user behavior in the real website,
- provide a fake HTTPS wrapper that would force the browser to display the SSL/TLS padlock (or other security indicator),
- prevent the user from viewing the HTML source code,
- load images and HTML code in the background for later use, or
- imitate the functionality of the browser toolbar (if it is overlaid with a graphical representation in order to hide the actual location of the content) in combination with client-side scripting software.

Hidden frames can also hide the address of the phisher's content server. Only the URL of the document containing the frameset will be accessible from the browser interface (e.g., from the location bar or the page properties dialog).

The use of DHTML allows the phisher to override the content of the legitimate site, effectively building a new site on top of the real page [46]. The DIV tag allows content to be placed within a virtual container, which can then be given an absolute position within the document. It can be positioned to obscure existing content with careful positioning. JavaScript can be used to dynamically generate the content. For example:

```
var d = document;
d.write('<DIV id="fake", style="position:absolute; left:200;
    top:200; z-index:2">
    TABLE width=500 height=1000 cellspacing=0
    cellpadding=14><TR>');
d.write('<TD colspan=2 bgcolor=\#FFFFFF valign=top  height=125');
```

This particular example uses JavaScript to generate the first few lines required to construct a DIV that will be positioned to obscure existing website content.

Users are increasingly aware of the visual clues that mark a secure and legitimate site [46]. For example: the https identifier at the beginning of the URL, the URL itself, the zone of the page source (e.g., My Computer, Trusted, Internet, etc.), and the padlock icon somewhere in the browser (indicating secure SSL/TLS communication). These visual clues can sometimes be difficult to mimic using traditional techniques; however, specially created graphics can be loaded and positioned over specific areas of the browser 'chrome' (the window frame, menus, toolbars, scroll bars and other widgets that comprise the browser user interface) using scripting languages.

For graphical substitution to be successful, the graphics must be consistent with the browser. It is trivial to detect the browser the user is using[5]; from this information, the correct graphics can be overlaid. Areas of interest for graphical overlays include [46]:

- location bar: altered to report the legitimate URL, rather than of the fake site (see Fig. 6);
- SSL/TLS indicator: a padlock is overlaid in the correct location to (falsely) indicate a secure connection;
- certificate details: fake details are displayed if a user reviews page properties or security settings, and
- zone settings (Microsoft Internet Explorer): this can be altered from "Restricted" or "Internet" to "Trusted."

The release of Microsoft Windows XP SP2 prevented Internet Explorer from being susceptible to some of the techniques for achieving these overlays, and other browser makers are following suit [43]. Alternatively, the location bar can be spoofed with JavaScript by [20]:

- Closing the actual location bar, and replacing it with a table. The first row of the table will contain the address bar (as an image), and the second row of the table will contain the rest of the page.

[5] This also allows the phishing scam to only focus on the users that use browsers with specific security vulnerabilities or that use browsers with specific functionality.

- Opening a small browser window that contains a white box with the legitimate website address inside; this window is then positioned over the browser's location bar.

Furthermore, the attacker can use JavaScript to create popup windows that display supplementary content. On arriving at the fake website, a phishing popup is created while the main browser is redirected to the legitimate site. This gives increased credibility to the popup window [20] (see Figs. 8 and 9).

Unlike Internet Explorer and other browsers, Mozilla and Firefox do not compile their graphical user interface into the browser itself. Instead, it is stored as XUL: XML User Interface Language. The XUL data for these browsers is readily available, and can be rendered inside the browser's content area. This could potentially allow a phisher to perfectly mimic the appearance of the browser, but allow them to arbitrarily set the location bar text or SSL/TLS padlock [43].

JavaScript can also be used to hijack inconspicuous events generated by the browser [43]. File upload controls can be embedded as form elements in website in order for phishers to retrieve specific files from the user; however, these elements cannot have default values. By attaching an event handler to the OnDragStart event (an Internet Explorer extension), the upload control can be appropriately populated if the user drags their mouse. Ensuring they do so is a task left up to social engineering. On the conclusion of the drag event, the form can be automatically submitted, along with the stolen file. Several attacks are known that work on the same basic principle, some of which are no longer possible after certain Microsoft security updates. Other exploits of this technique include inserting, and then activating, active content in a user's Favorites folder, inserting executable files into a user's start up folder, or installing a backdoor trojan (more specifically, installing Backdoor.Sokeven).

3.4 Server-side Exploits

Any discussion of the exploitation of server-side vulnerabilities to assist in a phishing attack quickly transcends phishing and enters the realm of general hacking and cracking; this would be somewhat beyond the scope of this chapter (see [34] for some additional details). Suffice to say there are numerous techniques for exploiting operating systems, applications and network protocols that a phisher could use if they were determined to comprise a legitimate website in order to conduct a phishing attack. However, two 'non-invasive' techniques of relevance to phishers will be discussed: cross site scripting and preset sessions [46].

Cross site scripting (CSS or XSS) seeks to inject custom URLs or code into a web-based application data field, and takes advantage of poorly developed systems [27]. Three techniques are typically used [46]:

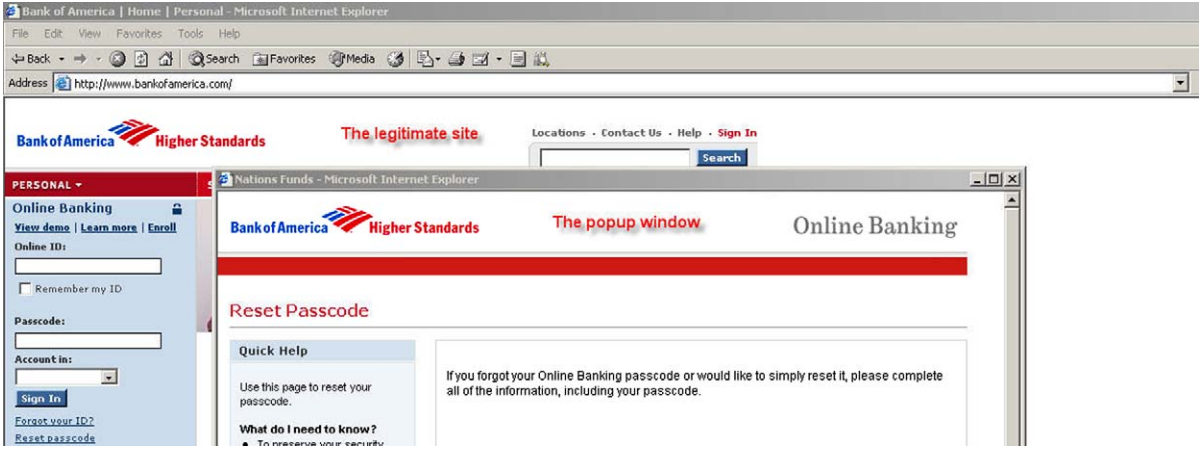


FIG. 8. This illustrates the use of a pop-up window over the legitimate site in the hopes of increasing the scheme's credibility. Obtained from the Anti-Phishing Working Group.

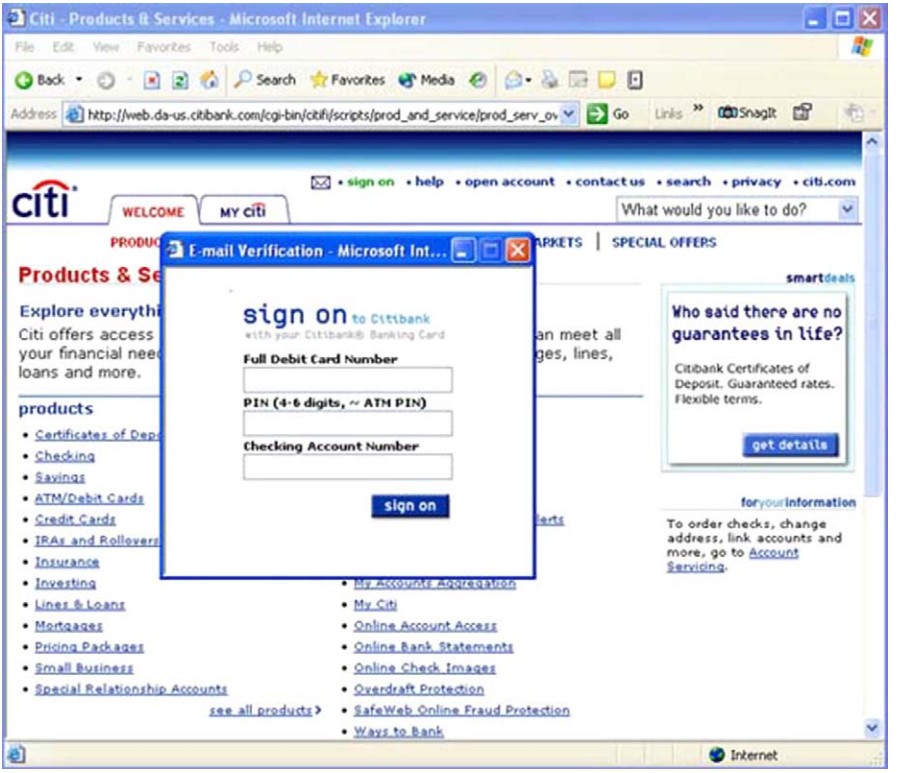


FIG. 9. This shows another pop-up window over a legitimate webpage. Using scripts, it opens up the real webpage and then opens a bare window popup asking for information.

- HTML substitution:

  `http://www.citibank.com/ebanking?URL=fakesite.com/login.htm`

  In this example, the standard legitimate website content is rendered, but the web application uses a parameter to identify where to load specific page content (for example the login box); in this case, that content is fetched from fakesite.com (whose URL could be obfuscated using previously described techniques).
- Forced loading of external scripts:

  `http://www.citibank.com/ebanking?page=1\&response=fakesite.com\%21secretScript.js\&go=2`

  In this example, a script to be executed is passed to the web application.
- Inline embedding of active content:

```
http://www.citibank.com/ebanking?page=1\&client=<SCRIPT>...
</SCRIPT>
```

In this example, the script is placed in the URL and executed by the web application.

Preset sessions use session identifiers. Session identifiers are typically used in HTTP and HTTPS transactions as a mechanism for tracking users through the website and to manage access to restricted resources (i.e., manage state). Session IDs can be implemented in a variety of ways; for example, cookies, hidden HTML fields or URL parameters. Most web applications allow the client to define the session ID. This allows the phisher to embed a session ID within the URL (that refers to the legitimate server) sent as part of the initial email [46]. For example,

```
https://mybank.com/ebanking?session=3V1L5e5510N
```

Once the email is sent, the phisher polls the legitimate server with the predefined session ID; once the user authenticates against the given session ID, the phisher will have access to all restricted content.

3.5 Client-side Vulnerabilities

Any discussion of client-side vulnerabilities is similar to that of its server-side counterpart: there are a multitude of vulnerabilities that a smart phisher could take advantage of in order to execute arbitrary code or to manipulate the browser. Given their exposure to the Internet, it is not surprising browsers suffer from a significant number of security vulnerabilities. Most browsers also support a number of plug-ins, each of which carries its own security risks. While patches are typically available in a timely manner, home users are notoriously poor at applying them quickly; therefore, phishers have ample time to exploit most security vulnerabilities, if they choose to do so.

Some past exploits used by phishers include [42,46]:

- Microsoft Internet Explorer URL mishandling: a URL such as:
  The real URL: `http://www.citibank.com%01@fakesite.com/phisher.html`
  What the user sees: `http://www.citibank.com`
  Where the browser goes: `http://fakesite.com/phisher.html`
  By inserting a %01 string in the username portion of the URL, the location bar displays `http://www.citibank.com`, while redirecting the user to `fakesite.com`. Earthlink, Citibank and PayPal were all targeted using this particular flaw.

- Microsoft Internet Explorer and Windows Media Player combination: this vulnerability allowed the execution of a Java JAR archive, disguised as a Windows Media Player skin, which could access local files.
- RealPlayer heap corruption: RealPlayer is available as a plug-in for most browsers, and allows the user to view the proprietary RealMedia format. By creating a malformed RealMedia file, and embedding it in a website to ensure it is automatically played, it is possible to cause a heap corruption, which would allow the execution of arbitrary code.

While malware can often be eliminated with a regularly updated antivirus utility, browser (or any client-side) exploits cannot be defended against until a patch is available and applied.

3.6 Context Aware Attacks

Context aware attacks [37] manipulate the victim into readily accepting the authenticity of any phishing emails they may receive. The first phase, which may involve interaction with the victim, will be innocuous and not request any sensitive information. Rather, the goal here is to ensure the victim will expect the message sent in the second phase. The second phase marks the dispatch of the actual phishing email; however, the email is expected by the victim, and therefore more likely to be considered authentic. The actions suggested in the phishing email would often arouse suspicion in the victim if viewed in isolation, but the preset context allows this to be avoided. Jakobsson [37] presented a context aware phishing scenario to 25 users, and recorded a 46% success ratio.

A simple example of a context aware attack involves targeting an eBay seller [37] (also see Fig. 10). Firstly, a seller is located who has an active auction and accepts payments via PayPal (but preferably not by credit card). At the end of the auction, a spoofed message is sent by the phisher from PayPal, indicating the successful buyer has paid for the goods won at the auction, but using a credit card (which the seller does not support). The email gives the seller two choices: either reject the payment, or upgrade their account to support credit card transactions; both these options require the seller to log into their account. By embedding an obfuscated URL to a fake website within the email, the phisher can easily record the seller's credentials. In this situation, the seller was expecting an email confirming payment; therefore, the spoofed email is expected, and is therefore viewed with less skepticism.

3.7 Empirical Results

Dhamija and Tygar [17] characterize the most common successful techniques employed by phishers. They reviewed the phishing attacks archived by the Anti-

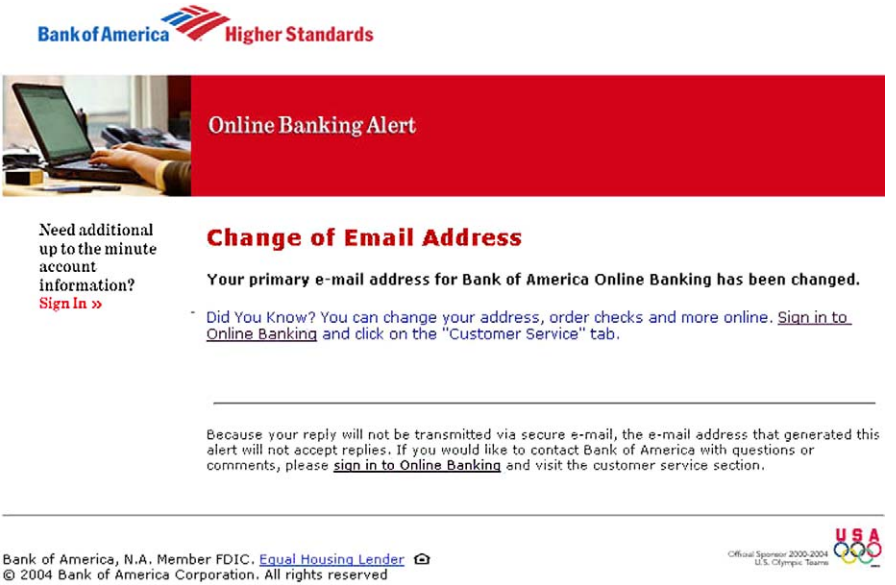


FIG. 10. This email is particularly well done, and illustrates a context-aware attack. On arriving at the site, the user is presented with a pop-up over the legitimate site, which gives the user the option of changing account details. It is not coercive and therefore not suspicious. They accept its legitimacy, as they require the ability to change their details. Obtained from the Anti-Phishing Working Group.

Phishing Working Group [4] over a period from September 2003 to mid 2005. Their findings were consistent with what is known about phishing: these attacks exploit human tendencies to trust certain brands and logos and that many phishing schemes prey on the widespread sense that the Internet is unsafe and that users must take the steps suggested by the attacker to ensure the security of their data. Furthermore, they concluded that the effectiveness of phishing schemes is raised when users cannot reliably verify security indicators. Unfortunately, this often the case, as browsers have generally not been designed with security usability in mind. More specifically, they identified the following phishing techniques as particularly serious:

- spoofed sender email addresses cannot be reliably detected,
- mimicked websites, with the same ‘look and feel’ as the legitimate site, cannot be reliably identified,
- obfuscated domain names are often undetected,
- images of URLs cannot be reliably distinguished from actual URLs,
- browser chrome cannot be reliably distinguished from web page content,

- images of legitimate security indicators (e.g., the padlock icon) can be mistaken for images of these icons,
- the meaning, and therefore the importance, of the SSL/TLS icon is not understood, nor is the concept of a certificate,
- the absence of security indicators is not reliably noticed, and
- multiple windows and their attributes cannot be reliably distinguished.

In related work, Friedman et al. [25] established that users found it difficult to determine whether a connection was secure under normal conditions. Intentional phishing and spoofing attempts will only make this task more difficult.

4. Anti-Phishing Techniques

The realm of phishing techniques is large and constantly expanding [16]; however, anti-phishing systems are not commonplace. Dhamija and Tygar [17] identify five basic principles that illustrate why designing secure interfaces is difficult:

1. Limited human skills: any security system design should begin by considering the strengths and weaknesses of the user, rather that starting from a traditional cryptography, 'what can we secure', point of view. For example, it has been shown [28] that users screen out commonly occurring notices (e.g., dialog boxes). Most browsers show such a warning when unencrypted information is submitted over the Internet; predictably, most users either ignore this message entirely or disable it.
2. General-purpose graphics: operating and windowing systems that allow general purpose graphics also permit spoofing. This has important implications for the design of spoof-resistant systems, as we must assume that the design can be easily copied.
3. Golden arches property: the marketing investment made by organizations in promoting their brand and visual identity is designed to invoke trust between the consumer and the organization. However, this trust can be abused: given principle number two, particular care must be taken to prevent the user from assigning trust exclusively based on graphics alone.
4. Unmotivated users: security is generally a secondary goal for a user conducting an online transaction; their focus will be on completing the primary goal (e.g., purchase a product online) rather than ensuring their security. In response to security warning like Fig. 11, most users just click "yes" without reading the warning message.

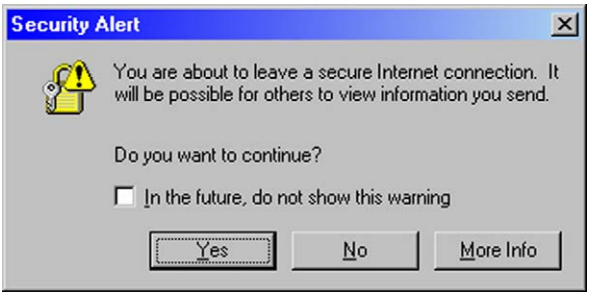


FIG. 11. Security warning pop-up message.

5. The 'barn door': once released, for whatever length of time, user information can be exploited. Secure systems should focus on protecting user information before they leave the user's control.

The authors argue that any complete phishing solution should fulfill all of these goals. In the following sections, we discuss available technical solutions for thwarting phishing attacks.

4.1 Detecting Phishing Attacks

Wenyin et al. [58] propose a system for detecting phishing website based on visual similarity. By examining the similarities between text, images, overall layout and overall style, an overall measure of similarity is produced. Experimental results indicate a low level of false positives based on a collection of 328 suspicious web pages. They intend the algorithm to be applied in a commercial setting by a monitoring company.

An automatic response to a phishing email can be used to detect the authenticity of the response [10]. It retrieves the embedded links in the email, visits the linked website, provides phantom user information, and analyzes the response from the fake website. If the visited website reacts differently from the expected behavior of a legitimate website, it determines that the site is a phishing site.

Some consideration should also be given to the structure of URLs over the entire website; simple URLs can be readily identified by users, and makes the identification of obfuscated URLs somewhat easier. Such updates to custom web applications can be done without interruption to users; however, secure application development requires skilled developers and thorough testing. The number of attack vectors available to the phisher can be substantially reduced through the use of these techniques,

and is relatively cost effective for the organization (when compared with the cost of an attack exploiting their web application).

Unicode attacks exploit the visual similarities between many Unicode characters (Section 2.3). Such attacks can be detected by character-character similarity and word-word similarity [26]. It has been demonstrated that this attack can be accomplished using the English alphabet, Chinese characters, or the Japanese alphabet.

4.2 Retaliation

Several anti-phishing companies offer retaliatory services [27]. They respond by sending phishing sites so much fake financial information that the sites cannot accept information from would-be victims. Most phishing sites run off of web servers installed on hijacked home computers and cannot handle much traffic. However, retaliatory services generally do not shut down phishing sites by overwhelming them with traffic, as occurs in a denial-of-service attack. They just send the sites as much traffic as they can handle and dilute their database with largely false information, a process known as *poisoning*.

Similar technique is proposed by [10]. Phantom user information is provided to the embedded website in the phishing email. By repeating this step rapidly, it can poison the phishing database.

4.3 Client-side Security Measures

The installation of generic security software on a user's local machine can circumvent a number of phishing attacks, in addition to protecting against a number of other security risks. Four key pieces of software should be installed on each user's machine:

- Anti-virus protection: removes malware and protects against the installation of new malware by phishers (and others). It should be regularly updated [22].
- Firewall/IDS: blocks unauthorized network connections that could indicate the installation of an unauthorized phishing program or use of a non-standard port for SSL traffic (which can indicate a phishing operation at work).
- Anti-spyware: removes spyware, which could potentially release sensitive user information to potentially malicious parties.
- Anti-spam: filters out unsolicited bulk email, including many phishing attacks.

Most consumers already recognize the value of anti-virus systems; it would be reasonable to assume they would be similarly interested in the Internet equivalents. While the purchase price for all four components can be substantial, well-regarded

freely available products are also available in each of the four categories. The combination of these services on a local machine can create some false positives; however, the net defense-in-depth effect gained positively impacts on a user's or an organization's security posture. Similar systems should also be deployed at the local network level [36].

Sophisticated email clients are widely used; however, only advanced corporate users require most of the functionality provided. The unnecessary functionality exposes the user to additional exploitable vulnerabilities that can be used by phishers [46]. The success of many phishing attacks can be attributed to the use of HTML email as it is particularly successful in obfuscating hyperlinks. By disabling HTML email in all email client applications, standard obfuscation and spoofing techniques can be rendered ineffective; however, this makes legitimate HTML emails difficult to read. The email client should also prevent the user from quickly executing dangerous content. At minimum, the user should be forced to save the attachment before opening it. This gives anti-virus software a better opportunity to consider the file, as well as preventing malicious code from compromising the rendering application (i.e., the email application). The use of simple clients, plain text email and automated attachment blocking can eliminate potential attack vectors for a phisher.

4.4 Web Browser Enhancement

Web browsers, when properly patched and configured, can be used as a defense mechanism against phishing attacks. In some respects they are similar to email clients: most browsers contain more functionality than the user will typically require [46]. The more functionality provided, the more security flaws are generally exposed. For example, in typical web browsing, a user will only use 5% of Microsoft Internet Explorer's functionality. Therefore, a browser that is appropriate to the user is important: simple web browsers are sufficient and more secure for most users who simply seek to browse the web.

Web browsers should also be properly configured to protect against phishing attacks. Popup windows should be disabled, along with native Java support, ActiveX support, and multimedia auto-playback and auto-execute extensions. In addition, non-secure cookies should not be stored, and new downloads should not be executable from inside the browser before being copied to a local directory.

The plug-in architecture provided by most browsers is being used to support an increasing number of anti-phishing systems. Security toolbars are widely in use. For example: Spoofstick [52] displays a website's real domain name; Netcraft Toolbar [45] displays a information about the site; Trustbar [33] displays a logos and the certificate authority of the website; eBay Account Guard [21] indicates the true eBay

site; SpoofGuard [11] calculates the 'spoof' score; and Web Wallet [63] tries to ensure users submit their data to the intended site. Typically these plug-ins are added to the browser toolbar, and confirm that the current URL is not part of a known phishing attack by contacting a centralized server. Ultimately, their effectiveness is dependent on the reporting mechanisms used by the system. Users often ignore toolbar messages and toolbars also make mistakes, so these toolbars must be used with caution [62]. The embedded links in phishing emails often contain a different link from that displayed in the text. For example,

```
<a href=http://123.132.234.87> http://www.goodsite.com</a>
```

An anti-phishing browser extension [40] would detect such discrepancy and warn the user. Other approaches to identifying phishing websites include [51]:

- Social networking: the toolbar informs the user if other people they know have viewed and trusted the site.
- 'Golden arches' property: the Trustbar takes advantage of the 'Golden arches' property described by Dhamija [16]. It displays the company's logo, as well as the logo of the company that signed the SSL/TLS server certificate (e.g., Verisign).

Dhamija and Tygar [17] propose the use of trusted security windows for the display and submission of credentials. The user would assign a unique security image as the background of the security window. The image would be stored locally, and could not be spoofed by a remote user. Therefore, the user would be aware when they were looking at legitimate security information or entering their username and password into an authentic form. The use of browser-generated random images or server-generated random images (also known as dynamic security skins) can also provide the user with a prominent visual indication of a secure connection. Z. Ye et al. [64] proposed trusted paths from the browser to the human user that might work under browser spoofing.

In the future, the anti-phish functions may become a built-in feature of web browsers. Netscape plans to release a web browser designed to resist phishing which will frequently update blacklists of suspected phishing websites and a whitelist of trusted sites. When a user follows an e-mail link and visits a trusted site, the browser will automatically render it, but block user access if the site is part of a known phishing attack. When a user visits a site not on a whitelist or blacklist, the browser renders it with enhanced security that disables ActiveX and JavaScript capabilities, which phishers could use to exploit vulnerabilities. Microsoft intends to provide anti-phishing functionality as a core part of its next browser, Internet Explorer 7.0. Deepnet Explorer 1.4 [15], a browser shell that uses the current version of Internet Explorer to render web pages, analyzes web addresses and warns users about those

on a blacklist of suspect sites. Users can then choose to either stop or continue trying to access a site.

4.5 Server-side Security Measures

Organizations should take a role in preparing users for an eventual phishing attack [24]. Most major online vendors, such as major banks, PayPal or eBay, already practice this to some extent (and to some effect) [46]. Communications from the organization should remind users not to release credentials to any other party, with an emphasis on prompting the user to consider the legitimacy of the motivation (e.g., email hyperlink) that drove them to the page. General phishing resources should also be made available to customers, detailing methods that can be used to ensure the validity of a site and how a customer can report a phishing scheme. Reported attacks should be responded to quickly, and users appropriately notified. Finally, all outgoing communications should be standardized; this reduces the likelihood legitimate communications could be confused with a phishing attack. All of these suggestions have a low cost to the organization, but must be delivered in a consistent manner where the customer is not overloaded with information.

Organizations can take a number of steps to validate their email communications with their customers, in order to make phishing attacks more obvious [46]. Emails can be personalized with some personal information known only to trusted organizations, such as greeting the customer by name, or including the last few digits of their credit card. A trail of trust can be established if subsequent emails precisely reference previous communications. Digital signatures can also be used to securely sign emails [1,55]; however, this relies on the user to validate the signature. Specialist web applications can also be made available to users to check the email was in fact sent from the organization. In order for these techniques to deter a phishing attack, the user must be aware of their existence and actively look for them.

Poor development techniques can expose custom web applications to some phishing techniques, such as cross-site scripting or the inline embedding of custom content (as discussed previously). Some of the key security requirements for a custom web application include [46]:

- all submitted content should be validated,
- session identifiers should be carefully monitored,
- tightly control URL redirection services provided,
- ensure safeguards are present in the authentication process (e.g., two-stage logins, anti-key logging processes, or personalized content), and
- use image cycling (regularly change the name of images on the site to render fake websites, that link to the legitimate website, out-of-date).

4.6 Alternative Authentication

Two-factor authentication (e.g., username/password and a secure token) has been suggested [46] as a possible solution to phishing attacks. By making the password time-dependent (i.e., it can only be used once), the phisher is limited in their ability to subsequently connect to the server. This system combats simple eavesdropping and password guessing; however, it is not a complete solution to phishing attacks [50,54]. Attack techniques such as man-in-the-middle or the use of Trojan horses will not be stopped: man-in-the-middle will still grant the phisher access, and Trojan horses will allow the phisher access to subsequent sessions from that machine. Two-channel authentication[6] is similarly vulnerable to active phishing attacks, but would eliminate some phishing attack vectors.

Delayed password disclosure [51] requires the server to continuously authenticate itself with the user. After a user enters each character of their password, a predefined image selected by the user is displayed. The pattern of images would be difficult for a phishing website to mimic. Mutual authentication is also achieved by using server and client side certificates. However, this requires users to have their certificate with them in order to connect to their bank; this inconvenience will limit the use of this technology [13].

Sophisticated browser password management can also be used to circumvent phishing attacks [51]. If the user allows the browser to manage all passwords, and a domain name is associated with each password, a user's credentials will only be automatically entered at legitimate websites [31].

Bellovin [6] believes that new authentication mechanisms will fail until prior relationships can be adequately captured. The use of certificates, both in email and on websites, merely guarantees the sender/website owns that particular domain name. It does not guarantee that this is the same party that the user gave money or sensitive data to. He proposes a simple solution to illustrate this point: if users were provided with the bank's certificate when opening an account, the certificate could be used to authenticate bank email and websites. The certificate is bound to a previous legitimate transaction, rather than simply being bound to a name.

4.7 Email Security

By modifying existing spam email filtering approaches, phishing emails can be detected and filtered by analyzing their content. According to [35], 54 out of 3,370 spam emails intercepted were phishing emails. Phishing emails typically contained

[6] Two-channel authentication requires the user to authenticate over two different mediums. For example, part of the authentication would involve the bank sending a challenge via SMS, and the user replying via SMS.

text related to banks and auction sites. By checking the text and other email characteristics such as sender, domain, and links, they formulated a scoring system to identify and block phishing mails.

Digital signatures can be used to make it easier to check the identity of the sender and the integrity of the message. However, it is still possible for a phisher to send a message using an anonymous public/private key pair. There are two popular standards for digitally signed email, S/MIME and PGP, which are supported by most Internet mail clients.

5. Comprehensive Anti-Phishing Efforts

Van der Merwe et al. [44] identify five key counter-attack categories for users and organizations to consider:

1. Education: users should be equipped with the skills to identify, and avoid, potential phishing attacks [36]. To a certain extent, this approach has failed: the vast majority of email correspondence reminds users that the organization will NEVER ask them for their password. Despite these regular warnings, phishing attacks continue to succeed in doing exactly that. Ironically, many phishing emails also include similar warnings.
2. Preparation: the danger of a phishing attack should be recognized, and policies put in place to manage or respond to such attacks. Different authentication technologies should be assessed for potential vulnerabilities. This particular category is of more relevance to organizations.
3. Avoidance: steps can be taken to avoid becoming the target of a phishing attack. For example, the use of anti-spam systems to filter out phishing messages or the use of Verisign verification on secure websites.
4. Intervention: when those behind phishing attacks step forward to influence the outcome of the attack, their success will be entirely dependent on the user. This relies on category one: the user should stop to think before submitting any personal information over the Internet.
5. Treatment: after a phishing attack, systems must be able to recover, identify the extent of the damage, and contact the appropriate organizations to prevent the misuse of sensitive information.

In other words, phishing cannot be prevented just by technical means alone; rather, a comprehensive response is necessary.

5.1 User Vigilance and Education

The behavior of users targeted by phishing attacks has been studied extensively in [18,19,48]. [18] observed the responses of 22 participants and analyzed the results by sex, age, education level, hours using the computer, etc. The study did not find any of these factors made a significant difference in the susceptibility of the user to the attack. Somewhat shockingly, even in the best case scenario, when users expected spoofs to be present and were motivated to discover them, many users could not distinguish a legitimate website from a spoofed website. In fact, the best phishing site was able to fool more than 90% of participants. In [19], 57 participants were tested and found that people use various strategies to distinguish phishing websites; however, these techniques were not necessarily effective. In [48], a user education course was offered and found that the user-awareness was greatly improved. Individual users are the most essential piece in an anti-phishing effort and they must take an active role to avoid becoming a victim of a phishing attack. Users can take several simple steps to protect themselves and their privacy:

- If a user gets an email warning that their account will be shut down unless they reconfirm billing information, they should not reply.
- Never respond to HTML email with embedded submission forms (i.e., never enter information directly into an email).
- Never click on hyperlinks within email even if they look legitimate; instead, directly type in the URL in the web browser.[7]
- Avoid emailing personal and financial information.
- Do not email back to confirm account information. Instead, call or log on to the company's website.
- For sites that indicate they are secure, review the SSL certificates by clicking the lock icon. Call the company if any certificate warning messages are displayed when you log into the website.
- Review credit card and bank account for unauthorized charges.
- Report suspicious activity.
- Ensure software updates are applied in a timely manner.

5.2 Proactive Detection of Phishing Activities

Various companies offer monitoring services, which are aimed at the early detection and elimination of phishing attacks. For example [27]:

[7] This would still leave the user vulnerable to a DNS poisoning attack; however, it would defeat a significant percentage of phishing attacks, which rely on malformed or disguised URLs.

- Corillian monitors and evaluates suspicious traffic on weekends, when most phishers conduct reconnaissance. By analyzing web logs, they are able to identify patterns of possible phishing behavior, such as downloading and saving images, or linking to images from a remote site. The process of verifying stolen accounts can also be detected.
- NameProtect identifies phishing attacks by monitoring spam from many sources (e.g., honey pot accounts [34]) and by checking domain name registration records for newly registered domains with names similar to that of their client.
- Cyota provides account information to phishers. The accounts themselves are set up in order to observe the phishing and fraud process. This allows the organizations involved to learn more about the nature of the attack.

5.3 Reporting and Response

Early reporting of phishing schemes allows them to be shut down as soon as possible and also allows users to be provided with some warning (e.g., by the organization involved or through anti-phishing software) [49]. Major banks and e-commerce businesses generally have reporting forms as part of their website; the US Bank provides an email address to forward suspect emails to, while Citibank also lists recent scams with a link to each one. Independent groups, such as the Anti-Phishing Working Group, also maintain information regarding known phishing attacks. Digital Phishnet is an organization formed to fight phishing attacks. It combines the forces of nine of the top ten US banks and financial services providers, four of the top five ISPs and five digital commerce and technology companies. They cooperate with the FBI, Federal Trade Commission (FTC), US Secret Service and the US Postal Inspection Service, under the aegis of the FBI's Internet Crime Complaint Center (IC3) [41].

Once reported, law enforcement officials are responsible for shutting the website down, tracing the source of the emails, tracking stolen funds and prosecuting those responsible. In Australia, the Australian High Tech Crime Centre and the Australian Computer Emergency Response Team are responsible for pursuing reported phishing attacks [49]. The URL contained within the phishing email will be used in a DNS search to find the ISP responsible for hosting the attack. This information usually allows the website to be quickly shutdown; however this may not be the case if the ISP is overseas, or in an unfriendly country. A G8 taskforce, consisting of 37 member countries, has recently been established to combat computer crime, including phishing. Of the phishing attacks recorded in May 2006 [3], 34.1% were conducted from inside the US, 15% from China, 8.17% from Korea, 3.94% from France, 3.38% from Germany.

Through effective reporting, historical conceptions about the spread of phishing attacks are changing [29]. Rather than spreading in a disorganized wildfire pattern, researchers now believe phishing attacks originate from specific IP blocks. CipherTrust [12] believes most phishing attacks are likely to originate from fewer than 5,000 networks. Messages sent from sources that do not typically send legitimate email are candidates for subsequent analysis. The IP addresses contained in such emails can then be followed to check for phishing attacks. More research is likely to allow researchers to better characterize phishing attacks.

5.4 Legal

In the United States, Democratic Senator Patrick Leahy introduced the *Anti-Phishing Act of 2005* on February 28, 2005 [30]. It allows prison time of up to five years and fines of up to US $250,000 for people who design fake websites for the purposes of stealing money or credit card numbers. California passed an anti-phishing law in September 2005, permitting victims to seek recovery of actual damages or up to $500,000 for each violation, whichever is greater [32]. Other US states, including Texas, New Mexico and Arizona, have also passed an anti-phishing law.

Although not common, some phishers get arrested. A 45-year-old California man, Jeffrey Brett Goodin, was arrested in January 2006 and charged with operating an online phishing scheme that targeted America Online customers [47]. He was charged with wire fraud and unauthorized use of a credit card. Goodin is alleged to have sent e-mail messages to thousands of AOL users to entice them to visit fraudulent websites he set up to collect personal information. Another phisher was arrested in August 2005 in Iowa [57]. Jayson Harris was charged with 75 counts of wire fraud for allegedly stealing credit card numbers and personal information in a phishing scheme targeting Microsoft's MSN customers. Other countries have followed the lead of the U.S. by tracing and arresting phishers.

Companies are taking proactive approaches in cracking down the phishers. On March 31, 2005, Microsoft filed 117 federal lawsuits in the US District Court for the Western District of Washington. The lawsuits accuse phishers of using various methods to obtain passwords and confidential information. AOL reinforced its efforts against phishing in early 2006 with three lawsuits seeking a total of $18 million USD under the 2005 amendments to the Virginia Computer Crimes Act.

6. Conclusion

Much of the Internet's malicious user population[8] has historically been motivated by challenge, curiosity, rebellion, vandalism, and the desire for respect and power. Modern trends in phishing reveal a very different situation: criminals have adopted the well-developed and well-known techniques of malicious users and are exploiting Internet users with sophisticated phishing attacks. The concept of phishing has mutated significantly since its creation almost ten years ago. Modern phishers are financially motivated and likely to pursue their attacks more aggressively than the average cracker [53]. The influence of organized crime further supports the changing nature of crime on the Internet. Phishing is also being used target individual users in an attempt to gain access to specific resources [27].

However, the outlook is not entirely bleak: anti-virus, anti-spyware and anti-spam systems are continuing to evolve, as are Internet browsers. If organizations prepare well, remain vigilant and follow attack trends carefully, they can respond quickly and effectively with a range of techniques to defend their customer's data. If individuals take a responsibility for their protection and adopt a defense-in-depth approach, consisting of a comprehensive and complementary toolkit of software and education, they can defend themselves against the most sophisticated attacks. There is no simple solution, but active and aware users and organizations have the ability to form a strangle-hold on this ever-growing threat. Consider yourself warned!

Acknowledgements

Fragments of this chapter have been taken from Berghel, "Phishing Mongers and Posers" [7] with the permission of the publisher.

References

[1] Anti-Phishing Working Group, "Proposed solutions to address the threat of email spoofing scams", 2003.
[2] Anti-Phishing Working Group, "Origins of the Word "Phishing" ", 2005.
[3] Anti-Phishing Working Group, "Phishing activity trends report", May 2006.
[4] Anti-Phishing Working Group, "Phishing archive", 2005.
[5] Anti-Phishing Working Group, "What are phishing and pharming?", http://www.antiphishing.org, 2006.
[6] Bellovin S.M., "Spamming, phishing, authentication, and privacy", *Commun. ACM* **47** (12) (2004) 144.

[8] Hackers, crackers and script kiddies.

[7] Berghel H., "Phishing mongers and posers", *Commun. ACM* **48** (4) (2006) 21–25.
[8] Berghel H., Brajkovska N., "Wading into alternate data streams", *Commun. ACM* **47** (4) (2004) 21–27.
[9] CACM Staff, "News track", *Commun. ACM* **48** (2) (2005) 9–10.
[10] Chandrasekaran M., et al., "PHONEY: Mimicking user response to detect phishing attacks", in: *Proc. 2006 Int. Symposium of World of Wireless, Mobile and Multimedia*.
[11] Chou N., Ledesma R., Teraguchi Y., Mitchell J.C., "Client-side defense against web-based identity theft", in: *11th Annual Network and Distributed System Security Symposium*, 2004.
[12] CipherTrust, http://www.ciphertrust.com.
[13] Clayton R., "Insecure real-world authentication protocols (or why phishing is so profitable)", in: *Thirteenth Cambridge Protocols Workshop*, Sidney, Sussex, UK, 2005.
[14] Cloudmark, http://www.cloudmark.com.
[15] Deepnet Explorer Browser, http://www.deepnetexplorer.com.
[16] Dhamija R., "Detecting phishing attacks: A user task analysis", in: *Authentication for Humans: Designing and Evaluating Usable Security Systems*, PhD dissertation, School of Management Information Systems, UC Berkeley, 2005.
[17] Dhamija R., Tygar J.D., "The battle against phishing: Dynamic security skins", in: *ACM Symposium on Usable Security and Privacy*, 2005.
[18] Dhamija R., Tygar J.D., Hearst M., "Why phishing works", in: *CHI 2006*, April 22–27, 2006.
[19] Downs J., Holbrook M., Cranor L.F., "Decision strategies and susceptibility to phishing", in: *Symposium on Usable Privacy and Security (SOUPS)*, July 12–14, 2006, pp. 79–90.
[20] Drake C.E., Oliver J.J., Koontz E.J., "Anatomy of a phishing email", in: *Conference on Email and Anti-Spam*, Mountain View, CA, USA, 2004.
[21] eBay Toolbar and Account Guard, http://pages.ebay.com/help/confidence/account-guard.html.
[22] Federal Trade Commission, "How not to get hooked by a 'phishing' scam", 2005.
[23] Fernandez J.D., et al., "Computer forensics: a critical need in computer science programs", *J. Comput. Small Coll.* **20** (4) (2005) 315–322.
[24] Flinn S., Stoyles S., "Omnivore: Risk management through bidirectional transparency", in: *Proceedings of the 2004 workshop on New security paradigms*, ACM Press, Nova Scotia, Canada, 2005, pp. 97–105.
[25] Friedman B., et al., "Users' conceptions of web security: A comparative study", in: *ACM CHI: Conference on Human Factors in Computer Systems*, 2002.
[26] Fu A., et al., "The methodology and an application to fight against Unicode attacks", in: *Proc. SOUPS*, 2006.
[27] Geer D., "Security technologies go phishing", *Computer* **38** (6) (2005) 18–21.
[28] Good N., et al., "Stopping spyware at the gate: A user study of privacy, notice and spyware", in: *Symposium on Usable Security and Privacy*, 2005.
[29] Goth G., "Phishing attacks rising, but dollar losses down", *Security & Privacy Magazine, IEEE* **3** (1) (2005) 8.
[30] Grant Gross, "Anti-Phishing Act pushes for 5 years and $250,000 fine", http://www.thestandard.com/internetnews/001048.php, March 5, 2005.

[31] Halderman J.A., Waters B., Felten E.W., "A convenient method for securely managing passwords", in: *Proceedings of the 14th International Conference on World Wide Web*, ACM Press, Chiba, Japan, 2005, pp. 471–479.
[32] Haskins W., "California passes nation's first antiphishing law", http://www.newsfactor.com/news/California-Passes-First-Antiphishing-Law/story.xhtml?story_id=010000Z2F774, October 4, 2005.
[33] Herzberg A., Gbara A., "TrustBar: Protecting (even naïve) web users from spoofing and phishing attacks", http://www.cs.biu.ac.il/~herzbea/Papers/ecommerce/spoofing.htm, 2004.
[34] The Honeynet Project & Research Alliance, *Know Your Enemy: Phishing*, The Honeynet Project & Research Alliance, 2005.
[35] Inomata A., Rahman S., Okamoto T., Okamoto E., "A novel mail filtering method against phishing", in: *PACRIM*, 2005.
[36] Internet Security Systems, "Protect your business from phishing", 2005.
[37] Jakobsson M., "Modeling and preventing phishing attacks", in: *Financial Cryptography '05*, 2005.
[38] Keizer G., "Phishing costs nearly $1 billion", InformationWeek, 2005.
[39] Kerstein P., "How can we stop phishing and pharming scams?", in: *CSO*, July 19, 2005.
[40] Kirda E., Kruegel C., "Protecting users against phishing attacks with antiphishing", in: *COMPSAC*, 2005.
[41] Kuchinskas S., "Phish fighters form alliance", http://www.internetnews.com/bus-news/article.php/3445511, December 8, 2004.
[42] Levy E., "Criminals become tech savvy", *Security & Privacy Magazine, IEEE* **2** (2) (2004) 65–68.
[43] Levy E., "Interface illusions", *Security & Privacy Magazine, IEEE* **2** (6) (2004) 66–69.
[44] Merwe A.v.d., Looc M., Dabrowski M., "Characteristics and responsibilities involved in a Phishing attack", in: *Proc. of the 4th International Symposium on Information and Communication Technologies*, Trinity College Dublin, Cape Town, South Africa, 2005, pp. 249–254.
[45] Netcraft Toolbar, http://toolbar.netcraft.com.
[46] Ollmann G., "The phishing guide", NGS Software Insight Security Research, 2005.
[47] Roberts P., "California man arrested in AOL phishing scheme", http://www.eweek.com/article2/0.1895.1916273.00.asp, January 27, 2006.
[48] Robia S., Ragucci J., "Don't be a phish: Steps in user education", in: *ITiCSE '06*, June 26–28, 2006.
[49] Rudd B., *An Analysis of Phishing and Possible Mitigation Strategies*, SANS Institute, 2004.
[50] Schneier B., "Two-factor authentication: Too little, too late", *Commun. ACM* **48** (4) (2005) 136.
[51] Sinclair S., Smith S.W., "The TIPPI point: Toward trustworthy interfaces", *Security & Privacy Magazine, IEEE* **3** (4) (2005) 68–71.
[52] Spoofstick, http://www.spoofstick.com.
[53] Treese W., "The state of security on the internet", *netWorker* **8** (3) (2004) 13–15.
[54] Tuliani J., *The Future of Phishing*, Cryptomathic Ltd., 2004.

[55] Tumbleweed, *Using Digital Signatures to Secure Email and Stop Phishing Attacks (White Paper)*, Tumbleweed Communications, 2005.
[56] Varghese S., "Korgo worm takes phishing to a new level", Sydney Morning Herald, Sydney, 2004.
[57] Wagner J., "MSN billing phisher arrested", August 24, 2005; http://www.internetnews.com/security/article.php/3529746.
[58] Wenyin L., et al., "Detection of phishing webpages based on visual similarity", in: *Special Interest Tracks and Posters of the 14th International Conference on World Wide Web*, ACM Press, Chiba, Japan, 2005, pp. 1060–1061.
[59] Wikipedia, "Identity theft", Wikipedia, the free encyclopedia, 2005.
[60] Wikipedia, "Phishing", Wikipedia, the free encyclopedia, 2005.
[61] Wikipedia, "Social engineering (computer security)", Wikipedia, the free encyclopedia, 2005.
[62] Wu M., et al., "Do security toolbars actually prevent phishing attacks?", in: *Proc. CHI*, 2006.
[63] Wu M., et al., "Web wallet: Preventing phishing attacks by revealing user intentions", in: *SOUPS*, 2006.
[64] Ye Z., Smith S., Anthony D., "Trusted paths for browsers", *ACM Trans. Inform. System Security* **8** (2) (May 2005) 153–186.