

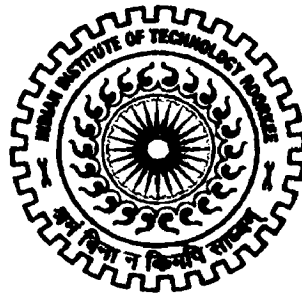
DESIGN AND IMPLEMENTATION OF AN ANTI PHISHING APPLICATION FOR THE END USER

A DISSERTATION

*Submitted in partial fulfillment of the
requirements for the award of the degree*
of
MASTER OF TECHNOLOGY
in
INFORMATION TECHNOLOGY

By

KAPIL OBEROI




**DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
ROORKEE -247 667 (INDIA)
JUNE, 2009**

Candidate's Declaration

I hereby declare that the work being presented in the dissertation report titled “**Design and Implementation of an Anti Phishing Application for the End User**” in partial fulfillment of the requirement for the award of the degree of Master of Technology in Information Technology, submitted in the Department of Electronics and Computer Engineering, Indian Institute of Technology Roorkee, is an authentic record of my own work carried out under the guidance of Dr. AK Sarje, Professor in Department of Electronics and Computer Engineering, Indian Institute of Technology Roorkee. I have not submitted the matter embodied in this dissertation report for the award of any other degree.

Dated: 15 Jun 09

Place: IIT Roorkee



(Kapil Oberoi)

Certificate

This is to certify that above statements made by the candidate are correct to the best of my knowledge and belief.

Dated: 15/6/2009

Place: IIT Roorkee.



Dr. AK Sarje

Professor.

Department of Electronics
& Computer Engineering,
IIT Roorkee,
Roorkee - 247667 (India)

ACKNOWLEDGEMENTS

At the outset I would like to thank Indian Institute of Technology, Roorkee and the Indian Army for giving me this opportunity to undergo the course for award of the degree of Master of Technology in Information Technology at this prestigious institute. It is my proud privilege and an honour to express thanks and profound gratitude to my supervisor Dr. AK Sarje, Professor for his invaluable guidance and constant motivation throughout the conduct of the dissertation work. I was able to complete this dissertation in time due to the constant motivation, guidance and support provided by him.

I am grateful to Mr. Nityam Parakh, my colleague and friend, for being an excellent peer and a ready reference for all my Java related troubles. His exceptional skills at Java programming helped me learn the language which was so necessary for my work. I am also thankful to all my other friends and colleagues who helped me directly and indirectly in completing this dissertation.

Most importantly, I would like to extend my deepest appreciation and love to my wife and daughter for their patience, love, encouragement and moral support which helped me bring this work to a logical conclusion.

(Kapil Oberoi)

ABSTRACT

Seeking sensitive user data in form of online banking username and passwords or credit card information through a combination of social engineering and technical subterfuge from unsuspecting Internet users, which may then be used by 'phishers' for their own personal gain is the primary objective of the phishing e-mails. With the increase in the online trading activities, there has been a phenomenal increase in the phishing scams which have now started achieving monstrous proportions, causing losses to the tune of billions of dollars worldwide.

In this dissertation we present an Anti-Phishing application designed to protect the end user from the threat of phishing attacks. A user is more likely to succumb to a phishing mail which apparently comes from an organization with which he has a relationship. Accordingly the application keeps track of the sites with which the user indulges in financial transactions and scans his e-mail account for mails which appear to have come from these institutions. Since the destination where a phishing email intends to lead the victim to is more dangerous than the email itself, we compare the source code of this destination web page against the source code of the home page and the login page of the institution the email claims to have come from. In case of a mismatch between the two pairs of source codes, the email is marked as a phishing email and a warning is generated about the same for the benefit of the user.

CONTENTS

Candidate's Declaration	i
Acknowledgements	ii
Abstract	iii
Contents	iv
List of figures	vi
List of tables	vii

Chapter	Topic	Page No
1.	Introduction	1
	1.1 Historical Background	1
	1.2 Current Status	1
	1.3 Reasons for Success of Phishing Attacks	2
	1.4 Motivation	3
	1.5 Problem Statement	4
	1.6 Organisation of the Report	4
2.	Phishing Attack Vectors	5
	2.1 Man-in-the-Middle Attacks	6
	2.2 URL Obfuscation Attacks	7
	2.3 Cross Site Scripting Attacks	9
	2.4 Preset Session Attack	10
	2.5 Hidden Attacks	11
	2.6 Context Aware Attacks	12
	2.7 Customer Data Observation	13
	2.8 Spear Phishing Attacks	13
	2.9 Pharming Attacks	14
	2.10 Vishing Attacks	14

Chapter	Topic	Page No
	2.11 In Session Phishing Attacks	15
3.	Defence Against Phishing and Related Work	17
	3.1 Enterprise Level Defence Mechanisms	17
	3.2 Server Side Defence Mechanisms	19
	3.3 Client Side Defence Mechanisms	22
	3.4 Related Work	25
4.	Anti Phishing Application : Design Details	29
	4.1 Background	29
	4.2 Design Principles	31
	4.3 Goals and Assumptions	32
	4.4 Main Functionality	33
	4.5 Handling Java Script Obfuscated URLs	38
	4.6 Connecting to the E-mail Server	38
5.	Implementation Details	40
	5.1 Modules Common to Both Versions of Application	40
	5.2 Modules Specific to Version 1 of Application	42
	5.3 Modules Specific to Version 2 of Application	43
6.	Results	43
	6.1 Experimental Setup	43
	6.2 Results: Anti Phishing Application Version 1	46
	6.3 Results: Anti Phishing Application Version 2	47
7.	Conclusions and Scope for Future Work	49
	7.1 Conclusions	49
	7.2 Suggestions for Future Work	49
	REFERENCES	51

List of Figures

Fig No.		Page No
Fig 1.1	Number of unique phishing sites from July 2006 to July 2007	2
Fig 2.1	Cross Site Scripting or CSS or XSS Attack	9
Fig 2.2	Preset Session Attack	10
Fig 2.3	Context Aware Attack	13
Fig 2.4	In Session Phishing Attack	16
Fig 4.1	Security warning pop-up message	30
Fig 4.2	Warning message to the user	34
Fig 4.3	Work Flow Chart of the Application (Version 1)	35
Fig 4.4	Flow Chart of Phase 1 of Anti Phishing Application Version 2	36
Fig 4.5	Flow Chart of Phase 2 of Anti Phishing Application Version 2	37
Fig 5.1	Table showing the data fed by the user	41
Fig 5.2	All_Recent_Mails table containing details of messages received since last check	41
Fig 5.3	Relevant_Mails table showing messages whose body is to be scanned	41
Fig 5.4	A Phishing Mail claiming to be from Axis Bank	42
Fig 5.5	Temp2 table showing the values of IP Addresses fetched from DNS	43
Fig 5.6	A sample view of tables containing the hash values of source code	44
Fig 6.1	Results chart for Application's Version 1	46
Fig 6.2	Results chart for Application's Version 2	48

To be able to commit crimes in complete anonymity having assumed the identity of unsuspecting individuals is the dream come true for any criminal. Prior to the advent and widespread usage of Internet, such efforts were confined to isolated individuals who were made to divulge with their sensitive information using 'social engineering' methods mostly over telephones. However, with the growing popularity and penetration of the Internet around the world ever since the early 1990's, the things have never been so easy for these criminals or 'phishers' (as this community is popularly known as).

1.1 Historical Background

The word 'phishing' initially emerged in 1990s. It describes the process of using lures to fish for sensitive user information. Although it is pronounced same as fishing, there are different theories as to where the "ph" in phishing comes from. It is argued that exchanging 'f' for 'ph' is a common hacker replacement (most likely an acknowledgement of the original term for hacking, known as 'phreaking'). The original form of hacking, known as phone phreaking, involved sending specific tones along a phone line that allowed users to manipulate phone switches. This allowed free long distance calls, or the billing of services to other accounts etc. There is also a view that 'ph' stands for 'password harvesting fishing'.

The first reported use of the term 'phishing' was in January 1996 [1] which reported the attacks on AOL accounts of the users. The attacks involved asking the users for their account details via emails or instant messages sent reportedly by AOL staff.

1.2 Current Status

Since first reported in the mid 1990's, the way phishers seek to trick the unsuspecting Internet users into visiting fake websites has gone through a sea change. Phishing scandals being reported in leading newspapers around the world have made the users wary of forwarding their sensitive account information through emails. The financial institutions too have taken upon themselves to educate their customers about this menace and constantly remind them not to divulge their account information to anyone either through the emails or telephones. This increased awareness amongst the users has forced the phishers to adopt new ways to try and get desired information out of the users, using the latest tools and technologies for same.

The preferred strategy of phishers today is to send out millions of spam mails to potential targets around the globe, masquerading as if these came from original institutions such as banks, insurance companies etc. These mails, while asking the users not to send any sensitive information via emails, urge the recipients to click on the embedded URLs which lead them to fraudulent but apparently official looking phishing websites where the gullible users are made to divulge with their personal information such as passwords, account numbers and such. These are then collected by the phishers from the server side using web tools such as key loggers.

As per the data collected by the Anti Phishing Working Group (APWG) [2] for the second half of year 2008, more than 15000 unique phishing sites were detected each month from Jul to Dec 2008 (as shown in Fig 1.1).

Unique Phishing Sites Detected : Jul - Dec 2008

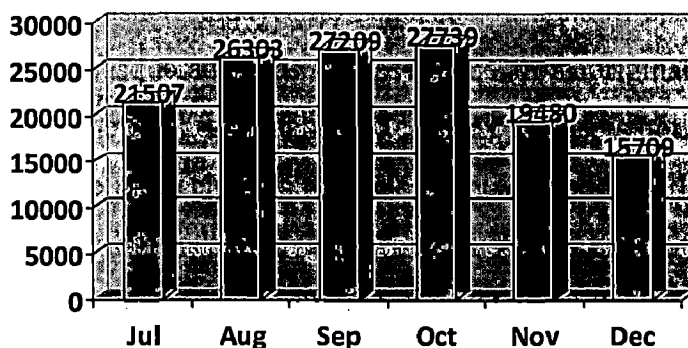


Fig. 1.1: Number of unique phishing sites from July 2008 to Dec 2008 (Data collected by APWG [2])

With the increasing awareness amongst the Internet users to guard against poorly composed emails containing generalised salutations and having numerous grammatical and spelling errors, phishers now pay a lot of attention to such minor details. Over the past few years, certain new variations of traditional phishing attacks too have been noticed [3]. In a particular variation (called spear phishing), personalized emails are sent to the users by phishers, who obtain such information from social networking sites. Pharming attacks which use techniques such as DNS hijacking, cache poisoning etc. are also being resorted to by the phishers.

1.3 Reasons for Success of Phishing Attacks

There is a growing awareness amongst Internet users today about the menace of phishing attacks owing to the widespread media coverage such attacks receive around the globe and also due to the efforts made by the financial service providers to educate their customers

about the same. Still we see that the number of phishing attacks have been on a rise with more and more people falling prey to the malicious designs of the phishers. In this subsection we try and analyse the reasons for the same.

What is essential for the success of a phishing attack is that the fake websites produced before the users must be as close to the original websites, in look, feel and content, as possible. Apart from this a few other factors too contribute to the success of the phishing attacks, which are listed below [4]:

- (a) A majority of the users of internet are naïve users, who do not have much information about the functioning of the internet, computer systems, secure delivery of web pages and the security indicators on the web browsers. They are thus perfect target candidates of phishing attacks.
- (b) Use of various URL obfuscation techniques by phishers such as use of similar looking domain names, Unicode encoding, using images to cover text and such, make it difficult for an ordinary internet user to detect a phishing attack.
- (c) The focus of the users towards the primary task of getting the transaction done over the internet causes his concern for security take a backseat, which in turn makes him more vulnerable to a phishing attack.
- (d) Lack of awareness on part of users that there are rouge elements over the internet, which are actively seeking our private information, makes them more prone to falling victim to a phishing attack.
- (e) Unawareness about the policies of financial institutions about seeking personal information of their clients over Internet is another reason for success of phishing attacks.
- (f) Technical advancements such as spam, DDOS and electronic surveillance have paid rich dividends to the phishing community in general.

1.4 Motivation

From what started as a relatively harmless and fun activity, phishing today has become a profound threat to online services provided by financial organizations, ISPs, retailers and governments. Almost all financial service providers' aim to provide Internet based services to their customers. The primary reason for this approach of the business organisations, apart from the obvious benefits to their customers, is that it offers them an excellent way to reduce the costs of their operations which translates into big profits for them. However the menace

of phishing emails is a big obstacle in their designs as it tends to keep their customers hesitant of net based transactions thereby forcing them to maintain significant physical presence to service them.

On an estimate, almost 5% recipients of phishing e-mails give away their personal information to these phishing sites while they are in operation [4]. A survey conducted by Gartner Inc., found that 3.6 million adults lost money due to phishing attacks during the period from Sep '06 to Aug '07, leading to a huge financial loss assumed to be of the tune of \$3.2 billion in US alone [5] compared to \$2 billion lost in the year 2006 [6]. This loss is not only due to the financial loss which is borne by the individuals and the financial institutions on account of the fraudulent transactions by the phishers. It is also due to the dent in the confidence and the resultant hesitancy of prospective clients of making use of web based businesses and services from the fear of being duped of their hard earned money. The magnitude of the problem is thus self evident and efforts are therefore underway to try and come up with a solution to this menace.

1.5 Problem Statement

In this dissertation work, we design and implement an Anti Phishing Application designed to protect naïve Internet users from the menace of phishing emails which seemingly have been sent by the organisations with which they have a transactional relationship.

1.6 Organisation of the Report

The organisation of this dissertation report is as follows:

In chapter 2 we discuss the various attack vectors employed by the phishers to try and lure their victims into divulging with their sensitive information. Chapter 3 discusses the various defence mechanisms against phishing to be employed at three logical layers viz Enterprise level, Server side and the Client side. We also discuss the work done in the field emphasizing the solutions proposed that seek to protect the end users from the phishing menace. The design details of our application are discussed in chapter 4 of this report. Chapter 5 covers the implementation details of the application. Chapter 6 gives the results achieved and finally chapter 7 concludes the dissertation work and gives suggestions for future work.

An attack vector may be defined as the path taken or the means adopted by a hacker/phisher to reach a destination computer [7]. Early phishing attacks mostly consisted of sending emails to unsuspecting individuals in order to get hold of their authorisation credentials. These emails generally claimed to be from the security branch of some legitimate institution warning the users' of some security breach associated with their accounts. They urged users to re-enter their passwords into the system by responding to the subject email, failing which their accounts were threatened to be blocked forever, causing them avoidable harassment. These emails were generally characterised as having poor grammar and numerous spelling errors and thus gave themselves away to a careful user. The threat of permanently deleting a user's account was used to prevent the user from reflecting over these errors and make him reply with the desired information before he could analyse the tell tale signs which gave away these phishing emails.

With the increased awareness amongst the users regarding phishing attacks, phishers too have responded with launching more sophisticated attacks to trick the users. Recent phishing emails have revealed that the phishers are much more careful with the grammatical and spelling errors in their emails making it almost impossible for a naive or even a sophisticated Internet user to judge the authenticity of an email based on its appearance. In line with the policy of financial institutions of not asking for authentication details through emails and using html forms for the same, the phishing mails too ask the receivers to do the same. They warn the users against forwarding authentication details using emails thus instilling in the users a false sense of security. The emails however ask the users to click on a hyperlink or a URL contained in the message body of the email. These obfuscated URLs redirect the users to fraudulent websites that are designed to look like the authentic website of the victim company by copying and using the bulk of the source code of the original site with slight variations to suite the phishers [8]. These sites are hosted on compromised home PCs or web servers and stay in operation for an average duration of 5 to 6 days.

Having gone through the effort of hosting a fake website on a compromised machine, a phisher is likely to be rewarded with his dues only if he is able to convince a user into clicking on the embedded URL which will actually lead him to that website. A number of methods have been devised by phishers to convince the users about genuineness of the URLs

and thus making them do what they want them to do. Some of the most commonly used methods to achieve this goal are described in subsequent subsections of this chapter. A few other variations of the phishing attacks which are being resorted to by the phishers of late are Pharming attacks, Spear phishing attacks, Vishing and In-Session Phishing attacks. These too would be discussed shortly.

2.1 Man-in-the-Middle Attack

In this form of attack, the attacker positions himself in between the user and the authentic server and proxy's entire communication between the two in real time. This attack is successful both for http and https communication. The user establishes a connection with the fake server (in case of https communication, the SSL connection is established between the customer and the proxy server) while the fake server connects to the authentic server (proxy server sets up its own SSL connection with the authentic server in case of https communication). The phishers is thus in a position to monitor the entire traffic flow between the user and the real server. For such attacks to be successful, the attacker must be in position to force the user to get connected to the proxy server and this he may achieve by:-

- **Transparent Proxies.** A transparent proxy, positioned either in the same network segment or on route to a server e.g. local ISP, can force all the network traffic (HTTP as well as HTTPS) to go through it and thus intercepts all the data.
- **DNS Cache Poisoning.** By replacing authentic IP addresses, for certain key domain names, with false ones, attackers can force the user to establish a connection with an authentic organisation via a server under their control.
- **URL Obfuscation.** Using a host of URL obfuscation techniques (as described in the succeeding paragraphs) the attacker may trick a user into logging onto their rouge server rather than the real server.
- **Browser Proxy Configuration.** By resetting the browser configurations of the user, an attacker may force victim's web traffic to be forwarded to a nominated rouge server. However unlike the above techniques, this method is not transparent to the users. In most of the cases the phisher would be required to physically change the browser settings before an attack may be launched.

2.2 URL Obfuscation Attacks

The success of a phishing attack hinges on the ability of the phisher to direct a user to a fake website without him getting suspicious about it. The attacker aims to achieve this goal by using certain methods of URL obfuscation wherein he tries to hide the actual destination of the clickable link behind an image or a character string which appears to the victim as the URL link to which he would be redirected. Some of the techniques employed by the phishers to achieve this aim are:

2.2.1 Bad Domain Names

It entails registration and use of bad domain names i.e. names that appear to be similar to the user as that of authentic websites, e.g. URLs `http://citibank.com` and `http://citibank.com.ch` appear to be the same to an ordinary user but point to different servers in reality. It is also possible that the phishers get their domain names registered in different languages, using different character sets which may appear same to an ordinary user (case in point is the use of “a” and “à” that have completely different meanings when used in a URL).

2.2.2 Friendly Login URLs

Many web browsers allow use of complex URLs that can include authentication information such as username and password. The supported format being: `URI://username:password@hostname/path`. The username and password fields of this format can be used by phishers to mislead the users to proxy web sites. Eg. Consider the URL `http://citibank.com:ebanking@citybank.ch/fake.html`. In this example `citibank.com` is the username; `e-banking` is the password, while the user is directed to the `fake.html` page of `citybank.ch`. However many current browsers have stopped supporting this URL encoding method.

2.2.3 Third Party Shortened URLs

URLs of many web based applications are very long and complex. This has given rise to many vendors who offer alternate shortened URLs, mostly free of charge. Phishers exploit such services by knowingly providing wrong or broken URLs in their messages to the users and also an alternate shortened URL to them with an aim to obfuscate the destination from the users.

2.2.4 Host Name Obfuscation

In order to navigate a website, the DNS translates the phonetic web address to the corresponding IP address. However it is possible for a phisher to use the IP address as part of URL to obfuscate the proxy server and hide the final destination from the end user. What compounds the problem even for a knowledgeable user is that there are variations of the classical dotted representation of IP addresses which may be used by the phishers. This can best be explained with the help of an example [9]. Consider a website whose URL is `www.mysite.com` and whose IP address is `210.134.161.35`. This address can then be represented in one of the following ways:-

- Using D-Word (Double word – 2 x binary words of 16 bits each, but expressed in decimal (base 10)) – `http://3532038435/`
- Using Octal (base 8) encoding – `http://0322.0206.0241.0043/`
- Using Hexadecimal (base 16) encoding – `http://0xD2.0x86.0xA1.0x23`
- In some cases it might also be possible to mix these formats

2.2.5 URL Obfuscation

Besides the techniques mentioned above, URL obfuscation can also be achieved in the following ways:-

(a) Escape encoding allows the inclusion of characters that may need special syntax in order to be correctly interpreted (e.g. a space in a URL string may indicate the end of the URL or it may be part of the URL). These are included as `%xx`, where `xx` is the hexadecimal ASCII code for the character. This also allows normal characters to be encoded in this way (e.g. `%41` is 'A' and `%20` is a space).

(b) Unicode encoding allows characters to be stored in multiple bytes. This permits a far greater number of characters (65,536) that can be encoded in comparison with ASCII (128), and allows a unique identifier for every character no matter what language or platform. In a Microsoft Windows environment, these characters can be encoded as `%u0000`, where `0000` is the hexadecimal code for the character (e.g. `%u0056` is 'V').

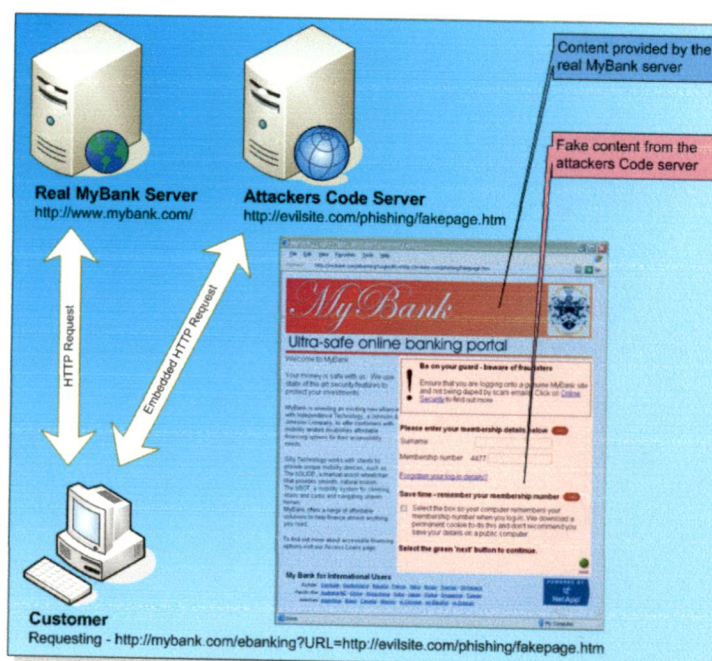
(c) UTF-8 encoding is a commonly used format of Unicode, and preserves the full ASCII character code range. This allows standard characters to be encoded (and obfuscated) in longer escape-coded sequences (for example, '.' can be encoded as %F8%80%80%80%AE).

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

2.3 Cross Site Scripting Attacks (CSS or XSS)

CSS attacks can occur in programs on websites that accept user input. These seek to inject custom URLs or code into a web-based application data field and take advantage of poorly developed web applications. If the program does not properly sanitise the input data, the vulnerable program may process input or even execute code that the original program was not intended to do. E.g. consider a URL that uses a vulnerable program on a legitimate e banking site <http://www.mybank.com/ebanking?URL=fakesite.com/login.htm> (refer figure 2.1).

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



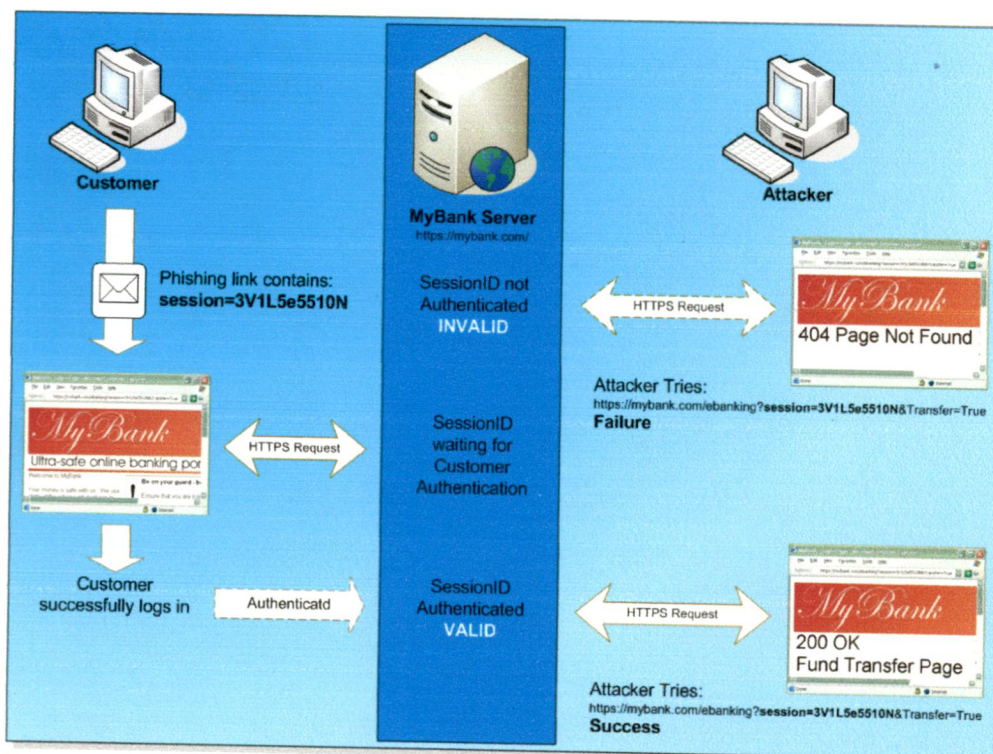
(Fig. 2.1: Cross Site Scripting or CSS or XSS Attack [9])

A CSS attack might also exploit vulnerable URL redirector programs. These are often used by websites to perform custom processing based on attributes such as web browser or authentication status. Phishers use these redirectors on legitimate websites to trick the users into visiting fake websites.

2.4 Preset Session Attack

As both http and https are stateless protocols, web based applications make use of session IDs to keep track of users accessing the applications and also their authentication. These session IDs may be implemented through cookies or fields contained within page URLs. Many poorly designed web based applications allow users to define the session ID which is then used for tracking the user as long as the connection is present. However, user authentication may still be required before granting access to the restricted content to the user. A phisher may exploit this in the following way:

The phisher will send bulk mails to the prospective victims containing a URL which has a predefined session ID and then wait for someone to respond. All this while the phisher too will keep banging on the authentic server with this session ID and will keep getting blocked for want of authentication. However his efforts will bear fruit as soon as a victim responds to the phishing email by clicking the URL link and authenticates himself at the application's server, from where on the attacker may take over (refer figure 2.2 below).



(Fig. 2.2: Preset Session Attack [9])

2.5 Hidden Attacks

In addition to the widely used URL obfuscation techniques by the phisher community, an attacker may make use of a host of scripting languages (such as HTML, DHTML or a host of other languages) which when interpreted by the victims' web browser will manipulate the display of information as desired by the attacker. The most common attack vectors used to achieve this aim include:-

2.5.1 Hidden Frames

Use of frames is a preferred method of delivering hidden content to a victims' browser window. Frames enjoy this support owing to their uniform browser support and easy coding style. A hidden frame may be designed to occupy 0% of the browser interface and reference the phisher's malicious contents. These can be used to:

- Deliver additional content such as overriding page contents or graphics.
- Executing screen grabbing / key logging observation code.
- Provide a fake secure https wrapper for sites content i.e. display a fake image of padlock at an appropriate location on the browser.
- Hiding HTML code from the customers.
- Loading images and HTML content in the background for later use by a malicious application.

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

2.5.2 Overriding Page Content

Use of DHTML allows the phisher to override the content of the legitimate site, effectively building a new site on top of the real page [9]. The DIV tag allows content to be placed within a virtual container, which can then be given an absolute position within the document. It can be positioned to obscure existing content with careful positioning. JavaScript can be used to dynamically generate the content, e.g.

```
var d = document;
```

```
d.write('<DIV id="fake", style="position:absolute; left:200; top:200; z-index:2">
```

```
<TABLE width=500 height=1000 cellpadding=14><TR>');
```

```
d.write('<TD colspan=2 bgcolor=#FFFFFF valign=top height=125');
```

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

2.5.3 Graphical Substitution

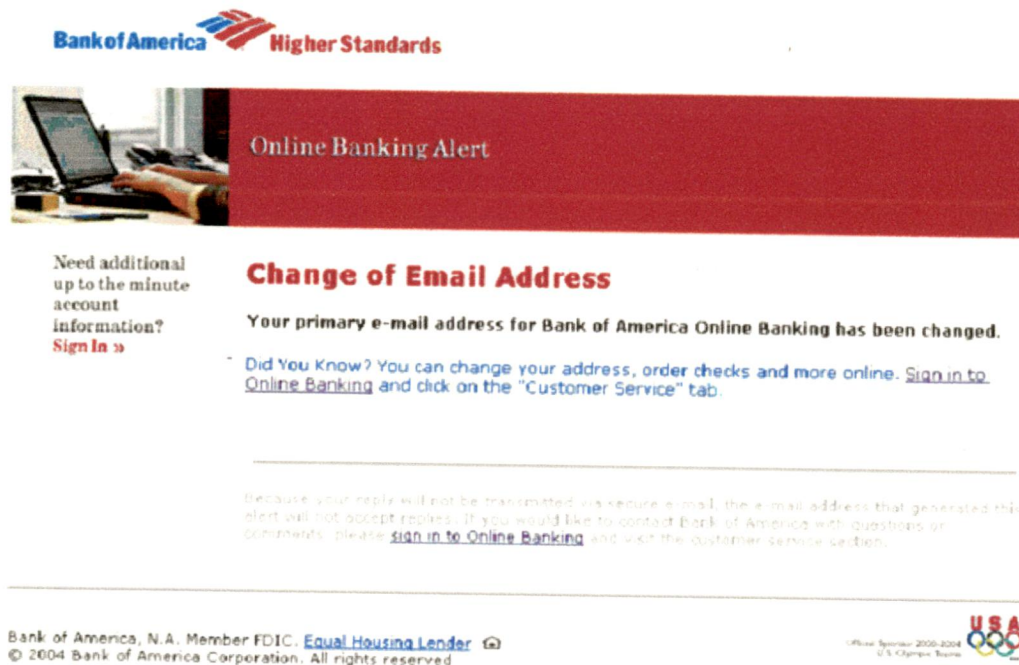
With increasing awareness amongst the users about the graphical signs used by browsers to point to the source of a phishing attack such as a padlock display to indicate secure https connections and the display of original URL in the browser URL field etc., the phishers use fake images and place them at the appropriate places within the browser 'chrome'. For graphical substitution to be successful, the graphics must be consistent with the browser. Phishers prepare the fake websites for a range of common browsers and once the users' browser is known, launch the appropriate webpage. Areas of interest for graphical overlays include:

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

2.6 Context Aware Attacks

These manipulate the victim into readily accepting the authenticity of any phishing emails they may receive. This form of attack is executed in two phases. The first phase is generally innocuous and the email will not request any sensitive information. Rather, it would make the prospective victim to expect the message sent in the second phase. The second phase marks the dispatch of the actual phishing email. Since the victim has been made to expect this email

in the first phase of attack, he is therefore more likely to consider the email as authentic. The actions suggested in the phishing email would often arouse suspicion in the victim if viewed in isolation, but the preset context allows this to be avoided. An example of this type of attack is illustrated with the help of an image below (figure 2.3):-



(Fig. 2.3: This email is particularly well done, and illustrates a context-aware attack. On arriving at the site, the user is presented with a pop-up over the legitimate site, which gives the user the option of changing account details. It is not coercive and therefore not suspicious. They accept its legitimacy, as they require the ability to change their details. [10])

2.7 Customer Data Observation

Phishers may also try to steal sensitive personal data by use of key loggers and screen grabbers. These are malicious software which a hacker is able to install on a compromised victim's machine. These malware are designed to keep a track of all the keystrokes that the user punches on his machine during a session or to take screenshots of the victims' computer every few seconds. The data so gathered may be collected through continuous streaming, local collection and batching of information which may then be uploaded on the phisher's server subsequently or by use of Trojan programs which allow the attacker to collect the user information as and when required.

2.8 Spear Phishing Attacks

This is a relatively new variant of the phishing attack vectors we have been talking of up till now. Unlike a classic phishing attack which operates by sending out millions of fake emails

to random Internet users and hoping that a few of the recipients would respond, spear phishing is a targeted e-mail attack that a scammer sends only to people within a small group, such as a company, government agency, organisation or group seeking unauthorised access to restricted data or for financial gain. Spear phishing attempts are not typically initiated by "random hackers" but are more likely to be conducted by "sophisticated groups out for financial gain, trade secrets or military information" [11]. Spear phishers often customize emails with information they've found on Web sites, blogs, or social networking sites like Orkut or Facebook. They also might create fake social networking login pages to lure people into sites where they're used to entering personal information.

2.9 Pharming Attacks

Pharming [12] is defined as a scamming practice in which malicious code is installed on a personal computer or server, misdirecting users to fraudulent Web sites without their knowledge or consent. Pharming attacks exploit the weaknesses inherent in the way a customer locates and gets connected to the online services of an organisation. These attacks manipulate the various components of core domain and host naming systems and misdirect the customer to an alternative destination which is under their complete control. Using techniques such as DNS hijacking, DNS spoofing and cache poisoning etc. pharmer have been successful in altering the DNS resolution information that is vital for the customers' browsers to locate and open the requested applications or files on the Internet. In another form of pharming attack, malicious code sent in an e-mail modifies local host files on a personal computer. The host files are used to convert URLs into the number strings that the computer uses to access web sites. A computer with a compromised host file will thus go to the fake web site even if a user types in the correct Internet address or clicks on an affected bookmark entry.

2.10 Vishing Attacks

This attack vector uses social engineering over telephone systems, using functionalities provided by VoIP, to gain access to private financial information from users for the purpose of financial gain [13]. The term is a combination of "voice" and phishing. Vishing exploits the public's trust in landline telephone services, which have traditionally terminated in physical locations which are known to the telephone company and associated with a bill-payer. The victim is often unaware that VoIP makes formerly difficult-to-abuse tools/features of caller ID spoofing, complex automated systems (IVR), low cost, and anonymity for the

bill-payer widely available. Vishing is typically used to steal credit card numbers or other information used in identity theft schemes from individuals. A typical vishing attack is launched as follows:

A hacker configures a war dialler to dial all the phone numbers in a particular locality. Once a connection is established, an automated recording using IVR warns the potential victim about fraudulent transactions on his credit card and asked to urgently contact a spoofed landline number. Placing his trust on the landline number provided to him, the user on dialling the number is greeted by an IVR machine which instructs him to enter his credit card/ account number, TPIN, PIN number, date of birth etc. and thus gathers all the information that is necessary to carry out a fraudulent transaction on the victims credit card.

2.11 In Session Phishing Attacks

This is one of the latest phishing attack vector that has been defined by the Trusteer [14] vide an advisory dated 29 Dec 2008. Unlike most of the attack vectors discussed so far, this method does not rely on the fact that its intended user is ignorant about phishing threat or that he is a naive Internet user. To this end this attack vector does not even make use of emails to lure the victims to visit the compromised/fake websites. This attack occurs when a user is logged onto an online banking application and performs some tasks. Once done, the user moves on to navigating other sites on other tabs/windows, leaving the banking site browser window/tab still open. A short time later a popup appears, allegedly from the banking website, which asks the user to retype their username and password because the session has expired, or complete a customer satisfaction survey, or participate in a promotion, etc. Since the user had recently logged onto the banking website, he/she will likely not suspect this popup is fraudulent and thus provide the requested details. The necessary conditions for success of this kind of attack are:

- (a) A compromised website which acts as a base for launching the attack.
- (b) The malware injected in the compromised website must be able to ascertain the websites which the victim is currently logged on to.

The first of the two conditions is comparatively easier to achieve, but the second one is comparatively difficult albeit not impossible. Security vulnerabilities have been located in the Javascript engines of all the leading web browsers (Internet Explorer, Firefox, Opera etc.) which allows a website to check if a user is logged on to another website. The injected

malware does not alter the appearance of the compromised website in any way and neither does it load the malware on the victim's computer. It is programmed to look for a website to which a user is logged on, from amongst a list of hundreds of websites, and generates a pop up message as shown in Fig 2.4 below.

This is a Security Alert you requested to help you protect your account.
Your account has been limited.
You have exceeded the number of three (3) failed login attempts.
To unlock your account, please login to your account
Thank you for your cooperation.
Regards,

(Fig. 2.4: In Session Phishing Attack)

Having seen the threat Phishing poses to e commerce activities it is imperative that the business community, particularly those involved in e finance activities, deploy a solution for the same. Accordingly, the research community has been roped in to try and come up with a possible solution to this problem. Also having seen a wide array of attack vectors available to the phishers (and new ones being constantly discovered and deployed), it must be appreciated that there is no single solution capable of taking on all the attack methods. So what we have is a game of one up man ship being played between the research community and the phishers with either side trying to gain the upper hand (the situation may be compared to one in between those involved in spreading computer viruses and the antivirus solution providers).

It is envisaged that preventing current and future phishing attacks is possible using a host of information system security technology and techniques deployed at three different logical layers [9]:

- (a) Enterprise Level – distributed technologies and third-party management services.
- (b) The Server-side – this includes the businesses’ Internet visible systems and custom applications.
- (c) The Client-side – this includes the users’ PC.

In this chapter we will briefly touch upon the defence mechanisms available at the Enterprise Level and the Server side, before covering in detail the defence mechanisms which need to be deployed at the client side as it forms the core area of work for this thesis work. We will also see some work done and solutions recommended by the researchers towards providing anti phishing protection to end users.

3.1 Enterprise Level Defence Mechanisms

At this level, the businesses and ISP’s need to implement solutions which in tandem with those employed at the server and the client side provide an in depth anti phishing protection to their users and their own employees. A few of the measures which need to be adopted at this level are:

- **Automatic Validation of Sending Email Server Addresses.** This essentially involves authenticating and validating the senders' e mail server by ensuring that its IP address is the authorised IP address for that particular domain. In case of a mismatch, the mail is dropped and does not reach its intended victim. Use of secure SMTP which entails exchange of certificates before a mail is transferred can be used to uniquely identify a trusted sender. In case it is discovered that the sender has a revoked, self signed or fake certificate, the delivery of emails from servers can be denied.
- **Digital Signing of Email Services.** The concept of digitally signed emails can prove to be an effective measure in the fight against phishing. Enterprises can configure their email servers to receive emails which are digitally signed. Alert messages may be attached along with emails which are unsigned in order to alert the recipients about the same. Similarly all the outbound email traffic from an enterprise's email server should be digitally signed. This would assure the receiving clients about the authenticity of an email and would help boost their confidence in the digital mode of transactions.
- **Monitoring of Corporate Domains and Notification of "Similar" Registrations.** Enterprises need to be on a lookout for domain names which are very similar to their own and may be used by fraudsters to deceive the customers of the organisation. In addition, if an organisation has multiple domain names, care should be taken to ensure that the same remain active at all times and do not lapse due to reasons such as non-payment of renewal fees. Many agencies allow registration of domain names which have lapsed and sell them to third parties who are then free to use them for their own end.
- **Perimeter or Gateway Protection Agents.** These agents when installed at the enterprise network perimeter can be used to monitor and control both the inbound and outbound communications. Typical enterprise level gateway services include Gateway Anti-Virus Scanning, Gateway Anti-Spam Filtering, and Gateway Content Filtering.
- **Third-Party Managed Services.** With a slew of attack vectors at their disposal and many more being discovered and deployed, phishers constantly develop methods capable of bypassing the perimeter protection agents. Third

party managed services play a vital role in improving the anti phishing security. While a particular enterprise may get only a handful of phishing messages, carefully worded so as not to trigger anti spam filters, managed parties see thousands of these as they provide their services to a host of organisations. They are thus in a position to see threats which would normally fall below the normal trigger threshold of perimeter spam filters as email volume is a key component in identifying malicious activities when dealing with phishing and spam mails. 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, remedial actions may be taken by the subscriber.

3.2 Server Side Defence Mechanisms

Developing intelligent and secure web applications for the organisations, which have the inbuilt ability to take care of a variety phishing attack vectors are crucial in protecting the users against a wide variety of phishing attacks. Stress needs to be laid on the security aspects during the development phase of the applications to ensure that there are no loopholes which can be exploited by the phishers to fool the applications and their intended users. Anti phishing protection at the server side can be provided by:

- **Improving Customer Awareness.** Educating their customers about the dangers of phishing attacks and the preventive steps they need to be aware of to stay clear of the same are an important measure in the fight against phishing. In addition the customers must be made aware of the policy adopted by the company in order to communicate securely with them via emails. A clear understanding of this policy will help the customers to identify the phishing mails if they encounter the same. Key steps in helping to ensure customer awareness and continued vigilance are:
 - Remind the customers repeatedly by posting small notifications on critical login pages about how the organisation communicates with the customers.

- Provide an easy method for customers to report phishing scams and also employ mechanisms to make sure that follow up action is taken on such submissions.
 - Provide adequate advice to the customers as to how they can verify the integrity of the website they are using.
 - Establish corporate communication policies and enforce them.
- **Providing Validation Information for Official Communications.** In addition to the steps taken by the organisation to educate their customers, following should be ensured:
 - Email communication with the customers should be personalised for that particular customer.
 - Quoting the reference to the previous correspondence on the subject should be resorted to as it plays an important role in establishing the trust in the communication.
 - Emails sent to the customers must be digitally signed and the recipients should be educated as to how they can validate the signatures.
 - Make available to the customers an portal on the corporate website wherein they can submit the messages they have received and verify the authenticity of the same.
 - The email can contain a personalised audio/video data which can act as a shared secret between the organisation and the customer.
 - **Ensure Securely Developed Web Application.** Organisations must make sure that their custom web based services are developed in such a way that they are able to neutralise a host of phishing attack vectors. Following points need to be kept in mind while developing such applications:
 - Need to ensure strict content validation protocols in respect of all data submitted by the users or other modules of the application. The content validation includes sanitising the input data and removing all the dangerous characters in the submitted data by decoding them so that the

same are not interpreted by the clients' browser in an ambiguous way and thereby providing an opportunity to the phisher to launch an attack.

- To neutralise threat of preset session attacks it must be ensured that the application URL does not contain the session information. In addition the session IDs must have expiry limits, any attempt on part of the user to submit an expired session ID must be redirected to the login page and a new session ID issued for the connection. Also the session ID provided to a customer over HTTP while in the process of establishing a HTTPS connection must be revoked and a new one issued once secure authentication is complete.
- The authentication process must at least be a two phase process. The first phase of authentication may consist of details which are not too secure, while in the second phase the users must authenticate themselves with two or more pieces of unique authentication information before they can access the application proper. To thwart use of key loggers by the phishers, use of facilities such as virtual keyboards and drop down menu boxes can be made use of.
- **Using Strong Token-Based Authentication Systems.** These involve use of external devices or software to generate a strong one time use password which is used to authenticate a transaction between the user and the organisation and which cannot be used for gaining repeated entry into the application. Bryan Parno et al. [15] propose the use of an additional authenticator based on a trusted external device such as a PDA or a mobile phone so that in order to gain access to a user account, the phisher needs to neutralise not only the authentication information in respect of users but also needs to compromise the external device which is in users' possession.
- **Keeping Naming Systems Simple and Understandable.** As already seen, a lot many attack vectors make use of the confusion caused by the organisations making use of complicated naming of host services and undecipherable URLs. In order to protect the customers, the organisations should as far as possible make sure the following:

- Always use the same root domain.
- Ensure automatic redirection of regional or other registered domain names to the main corporate domain.
- Use host names that represent the nature of web based application.
- Never keep session information in the URL format which may assist the phishers to launch preset session phishing attack.

3.3 Client Side Defence Mechanisms

Many information security experts hold the view that phishing is more of a user related problem than being a technical problem, their logic being that phishers get the information they require by tricking the users into giving it using social engineering tools. Speaking of average Internet/computer users, it can safely be said that they vary a great deal when it comes to comparing their computer related knowledge and skill sets. Providing an effective anti phishing security to the end users is therefore of prime importance in the fight against this bane of Internet because ultimately it is the end user who is the primary target of phishing attacks. Unlike the Enterprise Level and the Client Side security measures, which are dealt with by IT professionals, the end user cannot be expected to deal with the phishing emails without support from the IT industry. It is also a fact that the anti phishing defences are most poorly implemented at this end and a lot needs to be done to improve the existing state of affairs. Some of the measures which need to be adopted are:

- **Desktop Protection Technologies.** It is recommended that the users particularly those who spend any amount of time on the Internet, install some form of anti-virus protection on their machines. Accordingly most of the users of computers are aware of anti-virus solutions provided by one of the vendors such as Symantec, Macafee, AVG etc. Most of these products provide some or all of the services listed below:
 - **Anti-virus protection.** Removes existing malware and protects against installation of new malware by phishers (and others). It should be regularly updated.
 - **Personal firewall/IDS.** Blocks unauthorized network connections that could indicate the installation of an unauthorized phishing program or

use of a non-standard port for SSL traffic (which can indicate a phishing operation at work).

- **Personal anti spam.** Filters out unsolicited bulk email, including many phishing attacks.
- **Spyware detection.** Removes spyware, which could potentially release sensitive user information to potentially malicious parties.

Speaking in terms of anti phishing protection which these products are expected to provide to the users (in addition to their current functionalities such as auto updates of virus signatures, blocking and filtering spam email etc.) are:

- Detect and block attempts to install malicious software such as key loggers, Trojan horses and screen grabbers through emails, file downloads and scripted content.
 - Detect and block unauthorised out bound connections from installed software or active processes.
 - Ability to block inbound connections to restricted network ports and their services.
 - Detect anomalies in the network traffic profile and initiate appropriate counter measures.
 - Automatically block outbound delivery of sensitive information to suspected malicious websites.
- **Email Sophistication.** Email services nowadays provide to their users an ever increasing host of functionalities and sophistication, which are of little or no use to most of the users of these services. These unnecessary embedded services are exploited by phishers to launch their attacks. It is therefore recommended that these functionalities be turned off thereby rendering a vast majority of attack vectors employed by the phishers ineffective. Some of the most dangerous services which need to be blocked are:
 - **HTML Email.** As already seen in Chapter 2, HTML based emails are most preferred form of emails used by the phishers to launch phishing

attacks as these offer them the functionality for URL obfuscation, ability to embed scripting elements and automatic rendering of embedded multimedia content. Users must therefore configure their email clients to send and receive emails in plain text format. However disabling HTML based emails might cause certain emails to be rendered to the users in a format which may not be decipherable by an average user of email services. So while the visual appeal of the emails may be lost, the security quotient of the email service will be enhanced considerably.

- **Attachment Blocking.** Email service providers must make use of functionality which prevents the users from viewing/downloading the accompanied attachments directly. The attachments prior to being downloaded must be scanned for malicious content and must be blocked if the same is found.

- **Locking-Down Browser Capabilities.** Just as email services, modern web browsers come wrapped along with an impressive array of functionalities which apart from enriching our web browsing experiences, unintentionally provide gaping holes from the security point of view which can be exploited by the phishers. The following functionalities in a web browser need to be disabled to make more robust from the security point of view:
 - Window pop up functionality.
 - Java runtime support.
 - Active X support.
 - Multimedia and auto play extensions.
 - Prevent storage of non secure cookies.
 - Ensure that the browser cannot run any downloaded file directly. The same should be downloaded to a local disk and scanned by an anti virus program before being opened/executed.

- **Digital Signing and Validation of Email.** As seen already, it is possible to digitally sign emails using standard cryptographic techniques. Digitally signing

an email ensures the validity of the message content, ensuring that no one can alter it during transit. The users should therefore create their digital signature and make sure to sign all their outgoing mails. Similarly all the incoming emails should be checked for their digital signatures and the mails which are not digitally signed should be marked as suspicious and handled carefully.

- **General Security Awareness.** Besides the security aspects discussed above, the users need to be aware about certain general security issues to guard against phishing attacks. These are:
 - Never click on the embedded URL in an email which warns the user that his account will be blocked if he does not reply to the email at a very short notice. Users should check their account status by calling the call centre of the cited company or typing its web address which they know is genuine.
 - Never respond to HTML email with embedded forms, as any information sent using these forms over the Internet is sent in clear and can be monitored.
 - Be aware about the security indicators present in your web browser.
 - Avoid emailing sensitive financial information to anyone.
 - As a user one must be aware of the email policy of their service provider as almost all organisations do not solicit sensitive validation information using emails.
 - In case of https connections, review the SSL certificate of the web site and ensure that it has been issued by a trusted issuing authority.
 - A user must promptly check his account and credit card statements and in case any unusual transactions are noticed he must alert the service provider.

3.4 Related Work

Having realized the threat phishing poses to spread of e-commerce activities, researchers aptly backed by business houses around the globe are trying to come up with an effective anti

phishing solution. The research activities in this field have yielded some suggestions/solutions meant to be implemented at all the three logical layers. Since this dissertation work deals with an application designed to protect the end user from phishing attacks, we will be covering the work meant to bolster the client side defence mechanisms.

Rachna Dhamija et al. [16] proposed a scheme, Dynamic Security Skins, that allows the remote web server to authenticate itself using an abstract image and is difficult for the attackers to spoof. The scheme works in two phases. In the first phase, the browser extension provides an exclusive window dedicated to username and password entry. A dedicated image for each user is used to create a trusted path between the user and the window, thereby preventing spoofing of this window and the text fields. In the second phase, the remote web server generates an abstract image for each user and each transaction. This image creates a skin that automatically customizes the browser window. The scheme allows the user's browser to automatically generate the image which it expects to receive from the server. To authenticate the server, the user is just required to visually match the two images.

Engin Kirda et al. [17] have developed an anti phishing solution aptly called AntiPhish to guard users against a spoofed web site based phishing attack. The tool keeps track of the sensitive information of a user and generates warnings whenever sensitive information is typed on a form generated by a website that is considered untrustworthy. One of the drawbacks of the solution is that it lets the user go up to a stage where he is allowed to type in sensitive information on a form and then if the tool finds out that the website is untrustworthy; it warns the user against it. The user is thus susceptible to losing her sensitive data if the phisher employs tools such as a key-logger or a malware which is programmed to send screenshots of the user's console every few seconds.

Juan Chen et al. [18] have proposed a phishing site detection algorithm based on the characteristics of phishing hyperlinks. Having analyzed the phishing hyperlinks to have characteristics such as mismatch of visible and actual links, use of IP addresses in URLs, malicious coding of hyperlinks and use of DNS names that are similar looking and similar sounding to the authentic DNS names. The algorithm is based on the analysis of 203 phishing e-mail archives from 21 Sep 2003 to 04 Jul 2005 provided by APWG. Based on observed characteristics, the hyperlinks have been classified into five categories and the distribution of links in these categories is given in Table 3.1.

Category	Number of links
1	90
2	85
3	35
4	67
5	4
Total	281 **

(Table 3.1 Classification of Hyperlinks)

(** Note that a phishing hyperlink can belong to several categories at the same time.)

The algorithm is a rule based heuristic algorithm and works by analyzing the differences between the actual link and the visible link; comparing the actual link with websites that are included in the black list /white list maintained by APWG; pattern matching; and similarity check. Based on this analysis it declares whether a given site is a phishing site, possible phishing site or a non phishing site.

Maher Aburrous et al. [19] have proposed a method of phishing website detection based on applying fuzzy logic modelling based on 27 characteristics and factors that are the hallmark of a phishing website. Linguistic variables (high, medium, low) are used to represent Key Phishing Characteristic Indicators and the related website phishing probability. Website phishing detection rate is determined based on six criteria: URL & Domain Identity, Security & Encryption, Source code & Java Script, Page style & Content, Web Address bar and Social human Factor. The 27 characteristics are divided amongst these criterions. Each of these criterion has certain weight associated with it and are further classified into 3 layers having sub weights 0.3, 0.4 and 0.3 for layers 1, 2 and 3 respectively. The prediction about the status of the website is based on the formula:-

Website Phishing Rating = $0.3 * \text{URL \& Domain Identity crisp [First layer]} + ((0.2 * \text{Security \& Encryption crisp}) + (0.2 * \text{Source Code \& Java script crisp})) [\text{Second layer}] + ((0.1 * \text{Page Style \& Contents crisp}) + (0.1 * \text{Web Address Bar crisp}) + (0.1 * \text{Social Human Factor crisp})) [\text{Third layer}]$

Chandrasekran M. et al. [20] have proposed a solution to detect phishing attacks by generating automatic response to phishing email and then subsequent analysis of the response from the website can be used to detect a phishing website. The automatic response retrieves the embedded links in the email, visits the linked web site, provides phantom user

information, and analyzes the response from the fake web site. The difference in the expected behaviour of an authentic website from that of a fake website, determines that the site is a phishing site.

A solution proposed by Yue Wang et al. [21] claims that as average users visit only a small number of websites where they carry out financial transactions, it would be far more efficient to maintain a white list of these websites. They extended a web browser to maintain and manage a user white list which is created and updated by the user. Any attempt on part of the users to send sensitive information to a website not listed in the application is blocked by the application.

In addition to the solutions suggested by the research communities, anti phishing protection in some form or the other is also being provided by our web browsers. The latest web browsers come equipped with anti phishing solutions wherein they maintain a list of black listed sites which are confirmed phishing sites. Every site which the user wishes to open is checked against this list and the operation blocked in case the site appears in the list. This however is at best a passive approach against phishing and provides no protection against newly created phishing sites. Also the quality of protection provided relies heavily on the quality of black list maintained by the browser.

4.1 Background

As we have already seen, phishers are willing to go the distance in order to deceive a naive Internet user into divulging his sensitive authentication details which can then be used by them for some form of financial fraud or criminal activity. Although studies indicate that up to 5% of Internet users who received phishing emails, did actually respond to these mails and gave away their authentication details, the actual number of victims may be much higher than indicated as many people either fail to report the phishing attacks or do not own up to the fact that they have fallen prey to phishing for reasons such as resignation, embarrassment, fear or cognitive dissonance. So the question which needs to be answered then is that what makes the bogus websites so credible that people get fooled by them?

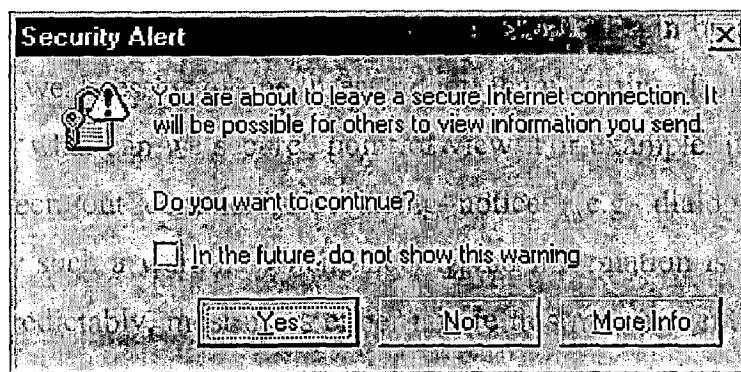
A careful analysis of this question reveals a very simple answer. For the success of phishing attacks it is imperative that the phishers are able to present websites to the potential victims which are as impressive as they are credible, forcing them to ignore the security indicators present in the web browsers. In addition it has also been shown that the users exhibit certain tendencies which inherently undermine security.

Rachna Dhamija et al. [4] carried out a study of level of common sense amongst the users with regards to their ability to locate a phishing site and which paints a rather dismal picture of a small sample of academic staff and students. A group of 22 participants were shown 20 websites and they were asked to determine which of those were fraudulent and why. The key findings of the study are summarized below:

- 90% of the participants were fooled by good phishing websites.
- Existing anti phishing security indicators such as browser address bar, security lock etc are ineffective, with many participants failing to notice them.
- Pop up warnings about fraudulent certificates were ineffective with a majority of respondents going ahead with the task at hand while ignoring the same.
- Participants irrespective of education, age, sex, prior experience and hours of computer usage proved vulnerable to phishing attacks.

It can thus be stated that designing a security product against phishing is not an easy task. A number of researchers have offered solutions to tackle this problem but have had limited success in countering the problem. Dhamija & Tygar [11] identify the following five basic principles that illustrate why designing secure interfaces is difficult:-

- **Limited Human Skills.** Security system design should begin by considering the strengths and weaknesses of the user, rather than starting from a traditional cryptography, 'what can we secure', point of view. For example, it has been seen that users screen out commonly occurring notices (e.g. dialog boxes). Most browsers show such a warning when unencrypted information is submitted over the internet; predictably, most users either ignore this message entirely or disable it.
- **Unmotivated User Property.** Security is generally a secondary goal for users conducting online transaction; their focus being on completing the primary goal (e.g. purchase a product online) rather than ensuring their security. In response to security warning like shown in figure 4.1 below, most users just click "yes" without reading the warning message.



(Fig. 4.1: Security warning pop-up message)

- **General-Purpose Graphics.** Operating and windowing systems that allow general purpose graphics also permit spoofing. This has important implications for the design of spoof-resistant systems, as we must assume that the design can be easily copied.
- **Golden Arches Property.** The marketing investment made by organizations in promoting their brand and visual identity is designed to invoke trust between the consumer and the organization. However, this trust can be

abused: given principle number three, particular care must be taken to prevent the user from assigning trust exclusively based on graphics alone.

- **The 'Barn Door' Property.** Once user information has been released on the Internet, for whatever length of time, it can be exploited by anyone who is interested in same. Secure systems should therefore focus on protecting user information before it leaves the user's control.

In addition it has been shown that users have a tendency to choose poor passwords and readily give it to complete strangers [15].

4.2 Design Principles

Based on the arguments given above, the principles which should guide the design of any anti phishing application meant to protect a user from phishing attacks are given below:

- As majority of Internet users are naive users, the application should not depend on the users input or feedback for its efficient functioning. However it is also true that any anti phishing application which seeks to protect users, has to have some input from them. There is thus a need to keep the desired input from the users to the minimum possible.
- As the users, irrespective of their computer knowledge levels are susceptible to be tricked into accepting a fake web site as an original, the decision regarding the authenticity of the website should not be left in their hands.
- Since users on their own are unable to authenticate the credential of the server with which they are communicating, the application should perform this task on their behalf to ensure that users' personal information does not fall into wrong hands.
- Presently the researchers are engaged in a ping-pong match with the phishers in which either side is trying to finish on top of the other. Researchers, it is felt are on a sticky ground in this respect as they are always on a defensive, trying to find an answer to every new trick the phishers come out with from their bag of goodies. There is thus a need to come out with a fundamental approach to counter the phishing threat.

4.3 Goals and Assumptions

Before we spell out the design of our anti phishing application, we would like to spell out the goals which we set out to achieve and the informed assumptions we made while developing this application. We would also like to mention that there are certain forms of phishing attacks, called Exploit based phishing attacks, which cannot be handled by an application of this nature. These attacks are more sophisticated and exploit some inherent weaknesses in the users' browsers or install some other malware such as a key-logger or a screen grabber which are programmed to keep tab of the user activity over his computer. The data so gathered may be collected by the phisher through continuous streaming, local collection and batching of information which may then be uploaded on the phisher's server subsequently or by use of Trojan programs which allow the attacker to collect the user information as and when required. To counter these attacks what is required is that these security issues be taken up by the respective browser manufacturing teams who should try and fix these security bugs. It is also essential that all the users are made available the latest security patches of their respective web browsers so that their systems are up to date. In addition the users should install the latest version of some antivirus/antispyware and regularly update its virus definitions to prevent phishers from installing malware on their computer machines. Our goals during the development of this application were:

- Protect the users against obfuscated URLs (misleading links) which might be present in their incoming emails.
- Protect the users against CSS and hidden attacks.
- Forewarn the user against presence of phishing emails in their email account, if found.
- Seeking minimum information from the user for running the application.

Having set the goals, we made the following assumptions while designing our application:

- A user is more likely to fall victim to a phishing attack if the phishing email received by her seemingly came from a financial/trading institution with which the user has a transaction relationship.

- For a naïve and inexperienced user, it would be better that the task to check the authenticity of the URLs embedded in the email be left to an application which cannot be fooled by the obfuscation techniques employed by the phishers.

4.4 Main Functionality

As brought out above, this application works on the premise that a user is more worried about the authenticity of emails which appear to have come from institutions with which he has a relationship, rather than emails received from all and sundry. As an example, consider a user who has an account with the ICICI Bank and not with Axis Bank. This user then is more likely to fall prey to a phishing mail which claims to be from ICICI Bank rather than a mail which asks him to verify his details with Axis Bank.

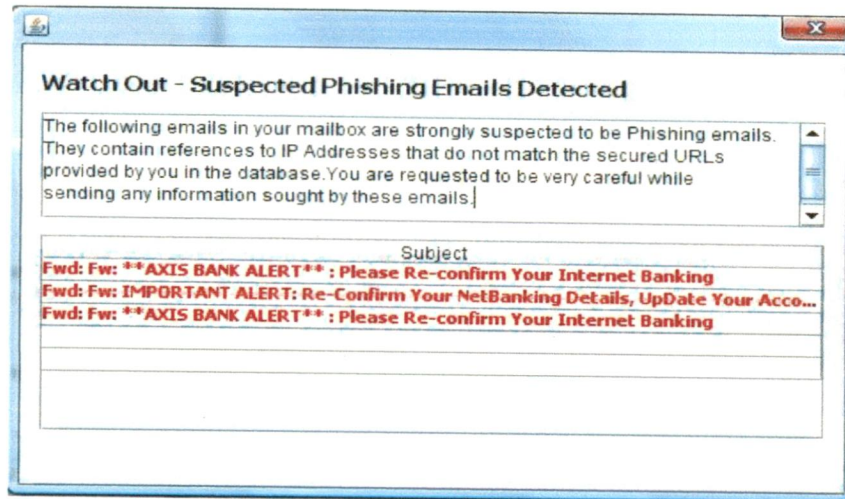
The first task at hand thus was to prepare a database of organisations with which users of the application had any form of relationship and which required authentication information in order to setup a connection. Subsequently the application establishes a connection with the 'Email Inbox' of the user, sorts the emails claiming from the organisations in the database and decide if the particular email was a phishing mail or not. During the course of our work we developed two versions of the application. In the earlier version, the IP address of the URL provided by the user was compared against those retrieved from the email messages to determine the genuineness of an email. In the later version, we use the digital fingerprints of the source code of web pages linked to these URLs to determine the authenticity of the emails. Given below are the details of both these versions along with the shortcomings noticed in the application's first version which prompted us to work towards the development of version 2 of the application.

4.4.1 Anti Phishing Module (Version 1)

To ascertain the websites which are of interest to a user, there is a need of user interaction with the application. Accordingly the application initially asks the user to enter the name and the trusted URLs of such institutions/websites where he sends his login details i.e. username and password. The application fetches the IP Address corresponding to this URL from the DNS server, calculates its message digest (using MD5) and stores this data in a table within the database.

On its subsequent run, the application connects to the e-mail service provider of the user (in this case Gmail from Google) and scans for URLs in the message bodies of only those

messages which appear/claim to have been sent by the websites of interest to the user. IP Addresses of URLs so located are fetched from the DNS server and their MD5 values calculated. These values are compared against the values stored in the database for that institution. A warning message which includes subject fields of all the e-mails which report a mismatch in the values of message digest are then displayed, cautioning the user that these mails are suspected to be phishing mails, as shown in figure 4.2 below.

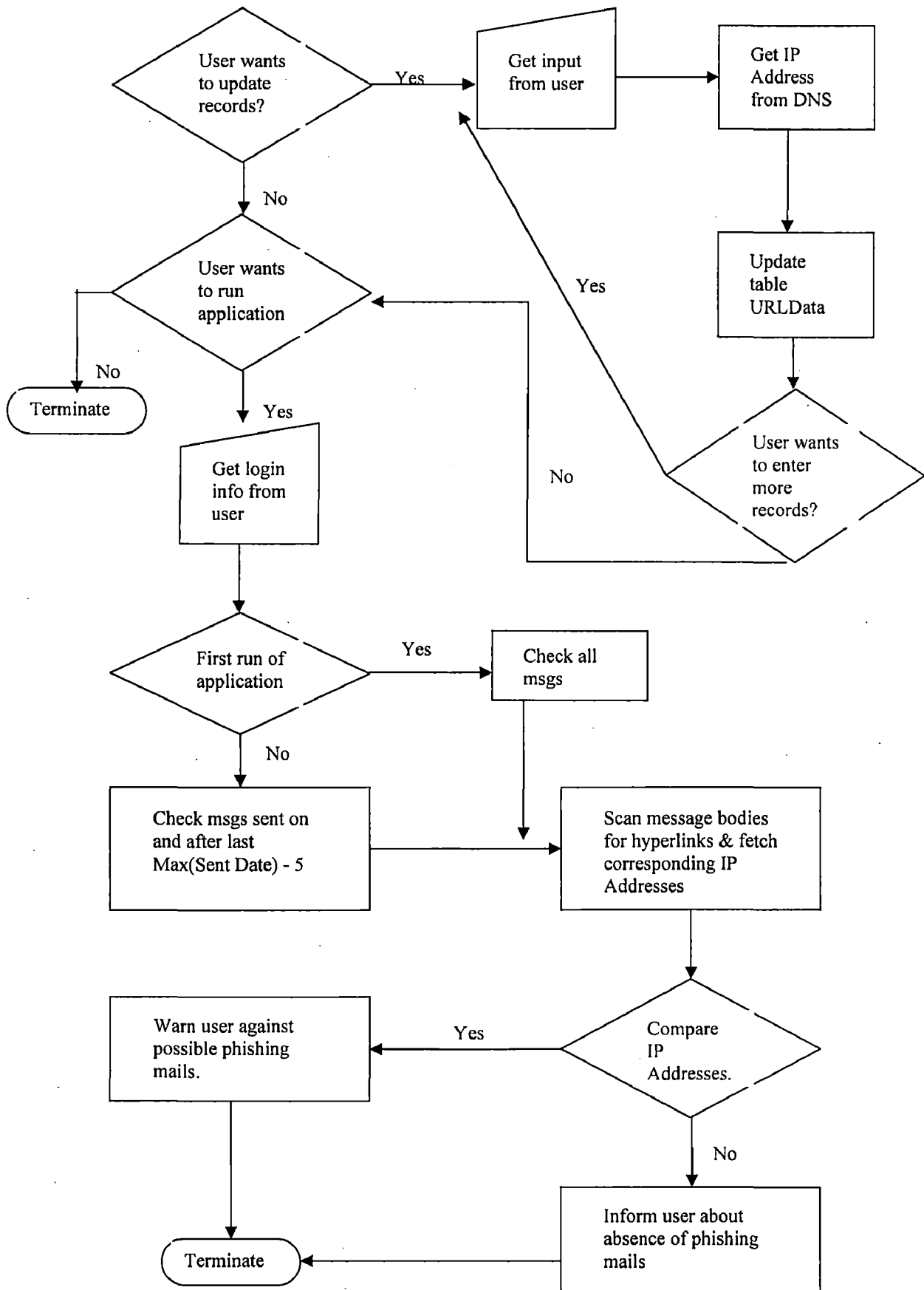


(Fig. 4.2: Warning message to the user)

The work flow diagram of this version of the application is shown in figure 4.3. Let us now talk about some of the shortcomings which were noticed in this version:

- Since the application depends on matching the IP address of the URL provided by the user to that of URL picked from the email message, the application was prone to give false positive results in case the organisation was using multiple servers or mirror sites for the purpose of load balancing.
- The application was not equipped to handle attacks wherein URL embedded in the email message was obfuscated using Javascript.
- CSS attacks and hidden attacks could go undetected because as seen earlier these attacks seek to insert extra code into the application while connecting the user to the authentic server of the service provider.
- Successful operation of the application was heavily dependent on the user provided input which in case of an ignorant user was prone to contain errors.

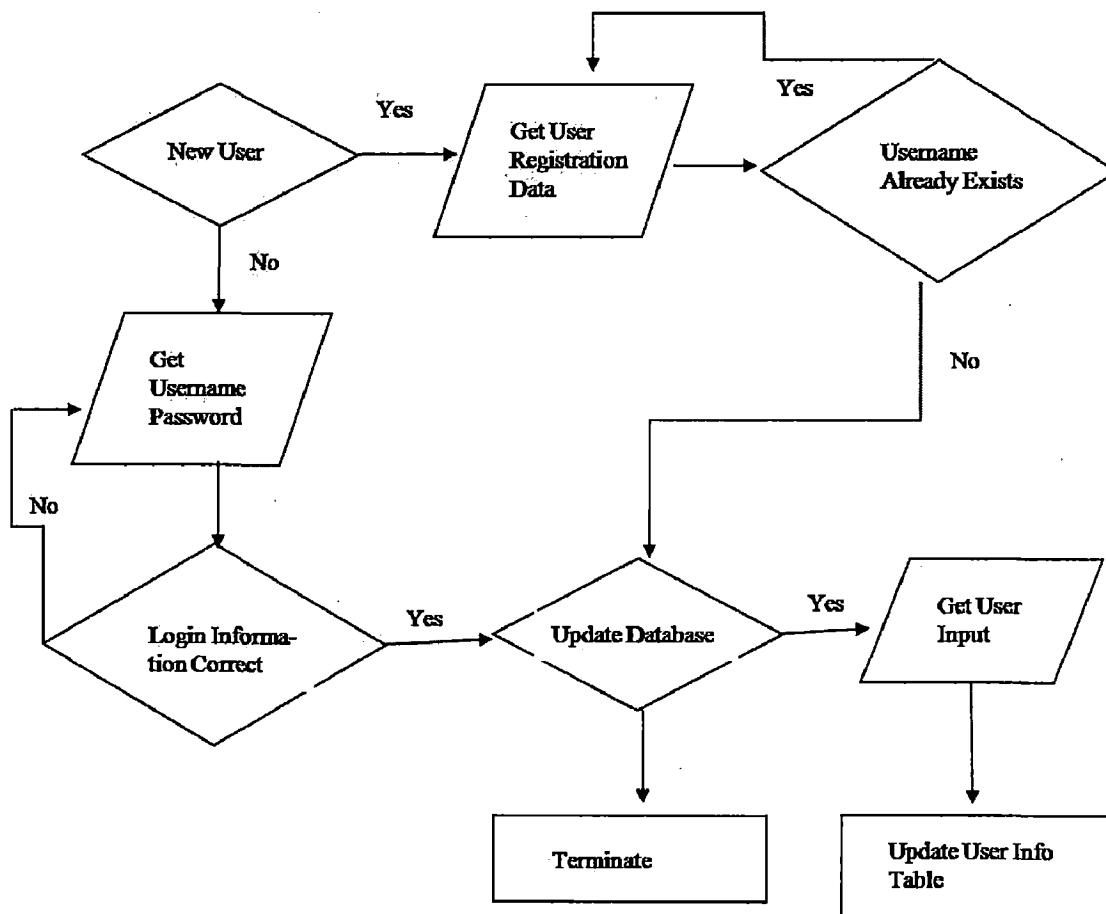
- Application had been developed to serve single user per computer and could not be used by anyone else (other than the first user) who wanted to check his email.



(Fig. 4.3: Work Flow Chart of the Application (Version 1))

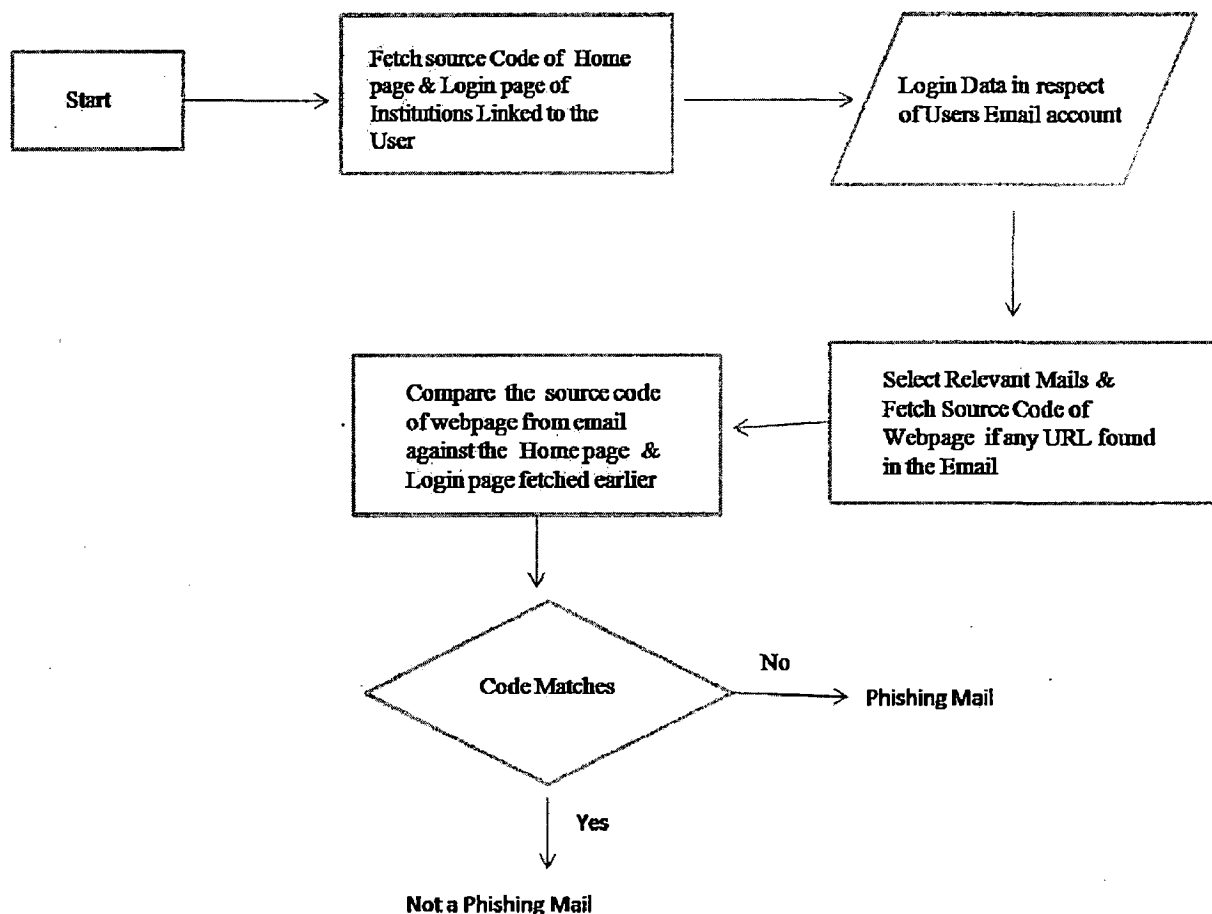
4.4.2 Anti Phishing Module (Version 2)

This version of the application supports its use by multiple users who can use it from the same machine. The users while running the application for the first time need to register themselves with it. During the registration process the users choose a username and password which they would require to log into the application when they run it. Having registered themselves, the users are asked to enter the data of the organisations with which they have an association. Since in majority of phishing attacks, the phishers pretend to be some organisation providing financial services to its customers such as a bank or an investment firm etc., we have provided a database of such organisations, which contains the URLs and IP addresses of the home page and the login page along with the name of the organisation. So the only data which a user needs to enter is the name of the organisation, thereby reducing the dependency of the application on the correctness of user provided data. The information about each user such as username (stored as its MD5 value), and the institutions with which he has a relationship are stored in the database. The work flow diagram of this portion of the application is given in figure 4.4 below.



(Fig 4.4: Flow Chart of Phase 1 of Anti Phishing Application Version 2)

When a user logs in to the application, it picks up the home page and login page URLs all the institutions related to the user and fetches the source code of these pages from the servers whose IP address are stored in the database. Using the previously stored IP addresses to fetch the source code ensures that in case the phisher has been successful in poisoning the DNS, it does not affect our application. The source code of all the pages fetched is converted into a string of characters (having removed all the blank spaces and null characters). MD5 values of characters of this string (1000 characters per batch) are calculated and stored in temporary tables in the database. The application then connects to the Gmail account of the user and selects the emails which seemingly came from the organisations of interest to the user. The application retrieves URLs embedded in these emails and fetches the source code of the pages they link to. MD5 values of the source code of these pages too are calculated in a similar fashion and stored in temporary tables of the database. The values of these tables are compared against the home page and login page tables of the particular organisation and in case of mismatch with either of these, the email is marked as a phishing email and a warning generated which is similar to one as shown in figure 4.2 above. The work flow diagram for this phase of the application is shown in figure 4.5.



(Fig. 4.5: Flow Chart of Phase 2 of Anti Phishing Application Version 2)

4.5 Handling Java Script Obfuscated URLs

Of all the obfuscation techniques employed to mask the actual URL in an email, the ones using Java Script are the most difficult to unravel. Almost all the anti phishing solutions suggested till date therefore recommend deactivating the Java Script functionality of the web browser to ward off such attacks. We however have tried to ward off such attacks in a different way. While scanning the message bodies of the emails, if we encounter a substring that goes like: “<a href = javascript:” we pick up the entire java script code from that email and run it in a separate frame of a browser window. The browser window is programmed to run two frames side by side. As the java script code executes in one of the frames, its location is picked by the adjoining frame. This frame now calls a servlet deployed on the local server, submitting the location of frame A to it. The servlet makes an entry of that URL in a temporary table of the database. The application now gets the URL from which to fetch the source code and proceed in the way already described above.

4.6 Connecting to the E-mail Server

POP3 (Post Office Protocol version 3) and IMAP4 (Internet Message Access Protocol) are the two most prevalent Internet standard protocols used by the local e-mail clients for e-mail retrieval from a remote server over a TCP/IP connection. Although IMAP4 is more user friendly and offers a host of facilities to the users, it is not supported by most of the ISPs (e.g. Yahoo mail does not support IMAP4). The wide popularity of the POP3 protocol is largely due to its appeal to ISPs, not to users. Using the POP3 protocol, ISPs can elect to not allow the user to leave a copy of the mail on the Mail Server, thus minimizing hard drive storage space.

Clients with a leave mail on server option generally use the POP3 UIDL (Unique Identification Listing) command. Most POP3 commands identify specific messages by their ordinal number on the mail server. This creates a problem for a client intending to leave messages on the server, since these message numbers may change from one connection to the server to another. For example if a mailbox contains five messages at last connect, and a different client then deletes message #3, the next connecting user will find the last two messages' numbers decremented by one. UIDL provides a mechanism to avoid these numbering issues. The server assigns a string of characters as a permanent and unique ID for the message. When a POP3-compatible e-mail client connects to the server, it can use the UIDL command to get the current mapping from these message IDs to the ordinal message

numbers. The present prototype of the Anti Phishing Module hence uses POP3 to connect to the Gmail account of the user to retrieve the desired information (Gmail incidentally supports both POP3 and IMAP4). POP3 works over a TCP/IP connection using TCP on network port 110, Gmail however uses the deprecated alternate-port method, which uses TCP port 995. One of the major disadvantages of using POP3 for mail retrieval is that it does not distinguish between a new message and a message which has already been fetched. As a result every call made to run the application results in it checking the e-mail inbox folder from the very beginning. Although this does not seem to be much of a problem in case there are a limited number of messages in the user's inbox, the time taken to complete run of the application increases manifold in case there are say a couple of thousand messages in the inbox folder (in our trial runs over an inbox containing about 300 messages, the application took an average of 6-7 minutes to give the result).

To get over the problem, this application makes use of the Sent Date field of the e-mail header (POP3 does not support Received Date field). The maximum value of the Sent Date field of all the messages checked is saved by the application and in the next run it starts checking the messages whose Sent Date is 5 days behind this maximum value. This is done because:

- It might so happen that due to some problem, the mails sent to the user's e-mail server get delayed and are not delivered by the time when the application is run. The mail clients typically try to send a blocked/undelivered mail to the intended recipients for 5 days and subsequently drop the email.
- The average life time of a phishing site is 5-6 days. Thus it may safely be assumed that if a mail is sent 5 days back and is yet to be delivered to the user it would have been rendered harmless by the time it arrives in her mailbox.

The project has been implemented in Java programming language using the Netbeans 6.0.1 Integrated Development Environment (IDE). The IDE is available as a free download from [22]. Besides the standard Java APIs, JavaMail API has been extensively used in the module. This API is an optional package which is used for reading, composing and sending e-mails. It provides protocol independent access for sending and receiving messages. To use the JavaMail API, there is a requirement to download the JavaMail implementation, unbundle the javamail-[version].zip file, and add the mail.jar file to the project. All versions of the JavaMail API require the JavaBeans Activation Framework, which is also required to be downloaded. The framework adds support for typing arbitrary blocks of data and handling it accordingly. After downloading the framework, it is required that we unbundle the jaf-[version].zip file, and add the activation.jar file to the project's CLASSPATH.

For the purpose of handling the data fed in by the user and that fetched from the mail server Microsoft Access Relational Database Management System (RDBMS) has been employed. The RDBMS comes along with the Microsoft Office Professional setup and is generally available in all the systems using Microsoft Office software suite.

We shall now take a look at the functioning of some important module of both the versions of the application. Since the second version of the application was developed to improve the functioning of the first version, there are certain modules whose general functionality is common for both, albeit with a minor variation here and there. Therefore we will first see the modules common to both the versions of the application and then see the functioning of modules specific to versions 1 and 2 respectively.

5.1 Modules Common to Both Versions of Application

DbDriver.java

This module is responsible for establishing a connection between the java programming platform and the Microsoft Access RDBMS. SQL queries from the application are passed to the RDBMS using this module and the result of the queries are passed back to the application as the desired result set for it to carry out further computations. At first run of version 1 of the application, the module checks to see if the URLData table consisting of data fed by the user

exists and creates the same if the table does not exist (refer figure 5.1). This table however is not used in the second version of the application.

URLData : Table				
Ser_No	Inst_Name	URL	IP_Address	Hash_Value
1	icici	www.icicibank.c	203.27.235.60	a6b6b224fdf79539de46cc4b1e07bec0
2	icicidirect	www.icicidirect.	203.27.235.20	f5fc1f816e34aacec06e2afd2cbfeab1
3	citibank	www.citibank.c	192.193.232.19	e93e8199c4d11ac51c52eb3ea4e93726
4	kapil	www.axisbank.c	210.210.17.218	c6e5ab32db0297da5961a945bd461695
5	team	mail.google.cor	209.85.153.83	c6b3b0f2cbdaa99dd565826b709fc4eb

(Fig. 5.1: Table showing the data fed by the user)

MailFetch.java

This module calls for a connection to the user's e-mail server. Having established the connection, the module calculates the value of Sent Date field which is then used to check the messages which have been sent on that date and after. Information regarding these messages is stored in All_Recent_Mails table (as shown in figure 5.2). The module then selects the messages which are relevant to the user by searching the From: header field for containing any substring as existing in the InstName field of the User_Inst_Data table. The results are stored in Relevent_Mails table (refer figure 5.3).

All_Recent_Mails				
Sender	Message_id	Date_Sent	Message_inde	Subject
abiciobee@in.com	<1243668514.70c767c26cb3143b>	30 May 2009	5	Reactivate your account
abiciobee@in.com	<1243798307.7a006957be65e608>	01 June 2009	7	This is PHISHING MAIL
abiciobee@in.com	<1243798318.812469e49663025b>	01 June 2009	8	This is not a phishing mail
abhdfcobee@rediffmail.com	<20090531201427.211114.qmail@>	01 June 2009	12	HDFC : This is a phishing mail
kapil.saachi@gmail.com	<3b2935f30905300035q597cb08e>	30 May 2009	6	Fwd: Fw: **AXIS BANK ALERT** : Ple
axisofpower09@yahoo.com	<580966.69794.qm@web111910>	01 June 2009	9	This is a phishing mail
axisofpower09@yahoo.com	<646724.39674.qm@web111919>	01 June 2009	11	Fw: Fwd: Fw: **AXIS BANK ALERT**

(Fig. 5.2: All_Recent_Mails table containing the details of messages received since last check)

Relevent_Mails				
Sender	Msg_Id	Msg_ind	Inst_Name	Subject
abiciobee@in.com	<1243668514.70c767c26cb314>	5	icici	Reactivate your account
abiciobee@in.com	<1243798307.7a006957be65e6e>	7	icici	This is PHISHING MAIL
abiciobee@in.com	<1243798318.812469e4966302>	8	icici	This is not a phishing mail

(Fig 5.3: Sample of Relevent_Mails table showing the messages whose body is to be scanned)

POP3.java

This module manages the task of connecting to the e-mail service provider using POP3 protocol, getting the 'INBOX' folder and opening it in the Read_Write mode. All the tasks related to retrieving the messages from the server are carried out by this module. The module is also responsible for disconnecting from the server once the application has completed its run.

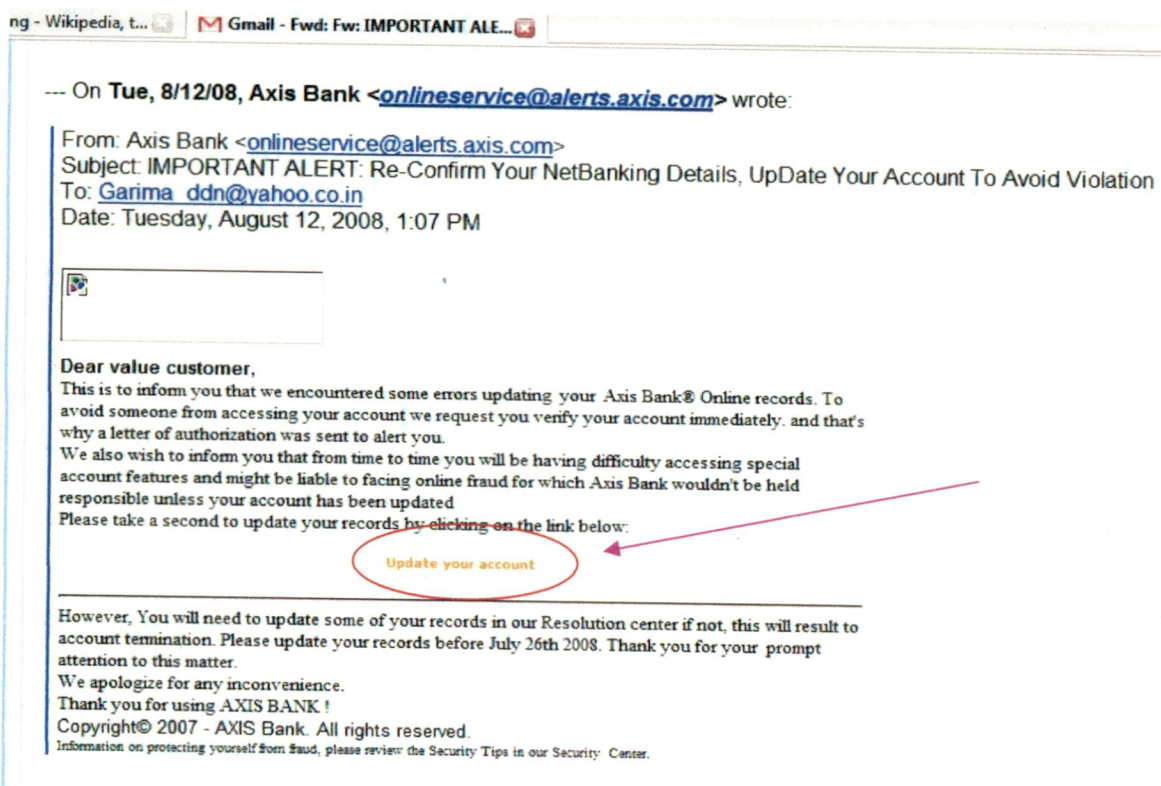
5.2 Modules Specific to Version 1 of Application

FetchURL.java

This module scans the entire message body (which is passed to it as a string) and searches for occurrences of hyperlinks which direct the user to other sites are mentioned. As an example, consider the e-mail (refer figure 5.4) in which the user is asked to click on a button to verify his details with the Axis Bank. The underlying code of the button however is:

```
<a href="http://www.era-info.es/components/com_expose/expose/swf/axisbank-security-update/1/axisbank.com/" rel="nofollow" target="_blank">
```

The code redirects the user to *http://www.era-info.es* instead of directing him to the website of Axis Bank. This module extracts this URL and fetches its IP Address from DNS. The retrieved information is stored in Temp2 table as shown in figure 5.5.



(Fig. 5.4: A Phishing Mail claiming to be from Axis Bank)

Temp2: Table	URL_Fetched	IP_Address	Inst_Name	Msg_Index	
	mail.google.com	209.85.153.83	team	1	Gmail is different. Here....
	mail.google.com	209.85.153.83	team	1	Gmail is different. Here....
	mail.google.com	209.85.153.83	team	1	Gmail is different. Here....
	mail.google.com	209.85.153.83	team	1	Gmail is different. Here....
	www.era-info.es	69.73.160.230	kapil	5	Fwd: Fw: **AXIS BANK ALEF
	www.riddim.com.br	72.55.148.240	kapil	3	Fwd: Fw: IMPORTANT ALER
	www.era-info.es	69.73.160.230	kapil	2	Fwd: Fw: **AXIS BANK ALEF
*				0	

(Fig. 5.5: Temp2 table showing the values of IP Addresses fetched from DNS)

FinalCheck.java

The comparison of the IP Addresses fetched by the FetchURL module against the corresponding values for a given Inst_Name as stored in URLData table is carried out in this module. In case a mismatch of the two values is detected, the Subject header of the relevant email is passed on for display to the user as suspected phishing mail, as shown in figure 7 of this report.

5.3 Modules Specific to Version 2 of Application

FetchSrcCode.java

This java class fetches the source code of the webpage whose URL is passed to it as an argument. The class contains a function named SrcCode which does the actual fetching of source code, removes the whitespaces from it and passes batches of 1000 characters at a time to the MsgDigest module of the application, which returns the hash value of the characters fed to it. These hash values are stored by this class into tables in the database whose naming convention is as follows:

- If the URL is of the genuine home page of the organisation, the table name would be 'InstName'homepage e.g. icicihomepage (refer figure 5.6).
- If the URL is of the genuine login page of the organisation, the table name would be 'InstName'loginpage e.g. icicilloginpage.
- If the URL is from the email, having some message index value (say 8), the table name would be 'InstName'8EmailURL e.g. icici8EmailURL (refer figure 5.6).

Count	HashValue
13	bd8845b5f766t
14	bb355dffbe45t
15	77485128e2e8t
16	5157d99f1bb2t
17	1c9d27f6a9a1d
18	e923c67988c87
19	fcf02f589f780a
20	114afaa39e544
21	664f98214bcf4
22	3645d602a4a91
23	06016d322feef
24	39a82fd846096
25	25d6e98f98cc1
26	8744ebbd210
27	0320e8630b4d
28	d8142a03f52dc
29	9641f09c90582
30	22bfe8a60189c
31	3e66276136d11
32	7f6df742973d9
33	e06a25a43a85f
34	e54af0d3a59f7

Record: 14 | 23 of 38

Count	HashValue
13	bd8845b5f766t
14	bb355dffbe45t
15	77485128e2e8t
16	5157d99f1bb2t
17	1c9d27f6a9a1d
18	e923c67988c87
19	fcf02f589f780a
20	114afaa39e544
21	664f98214bcf4
22	3645d602a4a91
23	f97e1542d847t
24	c4e3269853e61
25	8af4d63f91b7b
26	4ac7eb205dad
27	bd9203febaad
28	a042cb36fbb7f
29	a70b120c9f686
30	77e8a26c0cb22
31	d8e19f37a7f2c
32	28f26a81520bt
33	36489ec272d5t
34	ad09e5aa17d5t

Record: 14 | 23 of 37

(Fig. 5.6: A sample view of tables containing the hash values of source code)

FinalCheckSrcCode.java

This module checks the contents of 'InstName'EmailURL table against the contents of the InstNamehomepage and InstNameloginpage tables and in case a mismatch of values is found, the subject headers of the corresponding mails are passed on to the user as a phishing warning.

6.1 Experimental Setup

In order to test this application, we required a mixed dataset of phishing and authentic emails placed in an email account so that we could analyse the results generated. However, obtaining such a dataset of phishing emails proved to be a difficult proposition. APWG [2], an organisation set up to fight phishing maintains an archive of reported phishing attacks but does not part with it readily. We did approach them to grant us access to their archive of phishing emails but were denied the access on one pretext or the other. Searching for a ready database of phishing emails on the Internet, which could be used for our purpose, also yielded limited success. Finally we were able to download a corpus of 2279 phishing messages in mbox format from a website available at [23]. These messages had been collected over a period from Aug 2006 to Aug 2007. This dataset came with its own set of problems. Since it was in mbox format, we could view the dataset using windows Wordpad utility but could not use it to test our application because for that we required live emails in an email account. The dataset however proved useful in another way.

The relevant emails in the users' mailbox were selected based on a hypothesis that if the sender wants to trick the user into believing his assumed identity, he would use the name of the organisation (he is pretending to be from) in the From: header of the email. The dataset proved our hypothesis correct in that out of 2279 messages, 2058 messages were related to financial institutions and of these almost all the messages followed this rule.

In order to test our application we prepared our own dataset of about 100 phishing emails (claiming to be from Indian banks such as ICICI, Axis bank, HDFC bank etc.) which we downloaded from [24] and [25]. These messages were sent to the mailbox to be tested from email accounts specially created for this purpose i.e. these accounts contained the substrings icici or axisbank or hdfc in their addresses. Since the embedded URL links in these emails had long since been blocked, linking these URLs again to a fake webpage of these organisations was also required.

Jonathan Zdziarski et al. [26] have shown that in order to present a website which seems as authentic to the victims as it possibly could; the phishers often use the source code of the genuine websites, adding malicious code to it as suits their purpose. So we too used the

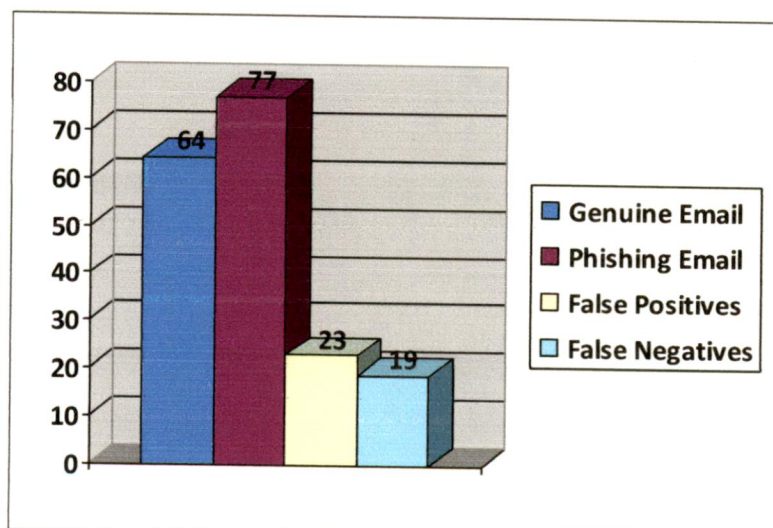
source code of the home page and the login page of the organisations suitably modifying them as needed for testing. These fake web pages were hosted on a local web server on our machine. The piece of code which we added to these pages alerted the user that they had reached a phishing page, whenever the page was loaded. The embedded URLs in the emails were modified and linked to the fake web pages, keeping the obfuscation technique used by the phisher in the original message as same.

6.2 Results: Anti Phishing Application Version 1

As brought out earlier, this version of the application used a mismatch in the IP addresses of the URLs provided by the user and those retrieved from the email as the parameter for decide whether an email was a genuine or a phishing email. When used to check a user’s mail account containing a mix of fake and genuine emails, it produced the following results:

- Total number of emails tested - 183
- Number of genuine emails - 87
- Number of phishing emails - 96
- Genuine emails declared as Phishing emails (False Positives) - 23 (26.43%)
- Phishing mails declared as genuine emails (False Negatives) - 19 (19.79%)

The graphical representation of the results is given in figure 6.1 below:



(Fig. 6.1: Results chart for Application’s Version 1)

Analysis of Results: Following conclusions can be drawn from the results achieved:

- This version of the application results in a high percentage of genuine emails being flagged as phishing emails. The reason for this anomaly being that in case the IP address of the URL of the organisation stored in the database did not match the query returned from the DNS while running the application, the email was marked as phishing email. The problem was more noticeable for large organisations such as Google, Citibank etc. which due to their sheer size have to maintain a lot of servers to manage their loads.
- A relatively higher percentage of phishing emails escaped the scrutiny of the application. Since CSS attacks and hidden attacks tend to take the user to the genuine server, with a part of the application being loaded from a different server when called for, these phishing emails were marked as genuine emails.

6.3 Results: Anti Phishing Application Version 2

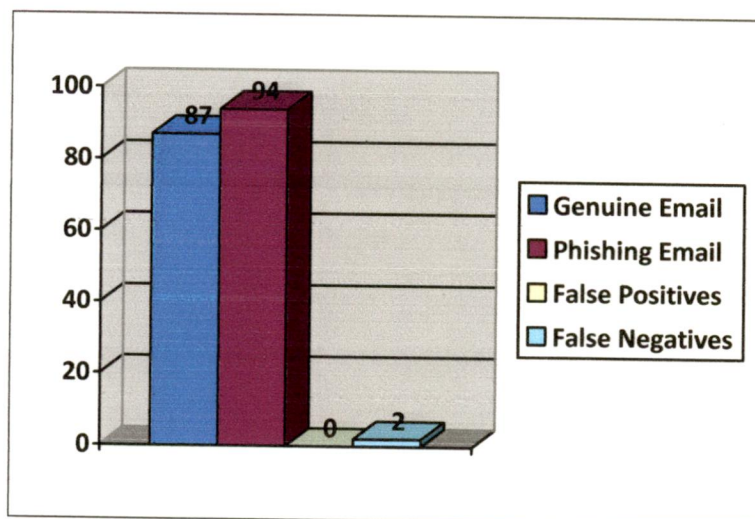
This version of the application looks out for a mismatch in the digital fingerprints of the source code of the web page retrieved from the URL embedded in the email message against the digital fingerprints of the home page or the login page of the institution provided by the user. When used to check a user's mail account containing a mix of fake and genuine emails, it produced the following results:

- Total number of emails tested - 183
- Number of genuine emails - 87
- Number of phishing emails - 96
- Genuine emails declared as Phishing emails - Nil
(False Positives)
- Phishing mails declared as genuine emails - 2 (2.08%)
(False Negatives)

The perceived improvement in the results obtained for this version of the application can be attributed to the following:

- Since the criterion for checking the authenticity of the email is the source code of the destination web page, rather than the link's IP address, no cases of false positive results were obtained as the source code of a webpage remains the same no matter which server it is hosted on.
- Reduction in the number of false negative cases can be attributed to successful prevention against CSS attack as they entail a change in the source code of the web page which also results in a change of their digital signatures.

The graphical representation of the results is given in figure 6.2 below:



(Fig. 6.2: Results chart for Application's Version 2)

7.1 Conclusions

In this dissertation we have tried to design an application which can provide protection to Internet users against phishing attacks. As majority of Internet users do not understand the way computers work and are not trained to detect the techniques employed by the phishers to lure them, they are susceptible to fall prey to the phishing attacks. However we feel that if the decision making power about the authenticity of an email message is handed over to a well developed computer application, many of the attack vectors employed by the phishers will lose their sting. Since the ultimate aim of the phishers is to lure victims to visit their fake websites, we can say that the intended destination of a phishing attack is much more dangerous than the phishing email which the phishers employ merely as bait to lure their victims. Also it is this intended destination which gives away the designs of the phishers.

We thus propose a novel approach to protect users from phishing attacks. As against a majority of solutions suggested so far which base their judgement about the authenticity of an email on the characteristic of its embedded URL, we compare the source code of the destination web page against the source code of the home page and the login page of the institution being supposedly represented by the phisher. The method we have employed has shown good results during its testing and we are sure that it can be used to counter phishing emails which even an experienced user may fail to detect.

The proposed solution also has the potential to ward off Pharming attacks. Since the application maintains its own list of IP addresses in respect of the institutions of interest to the user, any change in the DNS effected as a result of an attack by the phisher's may prove to be ineffective and may be caught before it can cause any damage. However there is a need to look into this aspect more thoroughly.

7.2 Suggestions for Future Work

Our work towards the development of an anti phishing application is a small step towards fighting this crime which is negatively affecting the growth of e commerce services because of the inherent fear of phishing amongst the Internet users. However we feel that work needs

to be done in the following areas in future to make this application more robust, easy to use and trustworthy:

- Application could be integrated with a web browser as a plug in so that users could use it with ease.
- Scope that this application is provided as an in built service by the email service providers could also be considered.
- Organisations such as APWG maintain a blacklist of confirmed phishing websites, which could be integrated with the application.
- A method could be devised to report the phishing sites caught by the application to the authorities and the organisation involved.
- The effectiveness of this application against Pharming attacks may be looked into.

REFERENCES

- [1] Hal Berghel, James Carpinter and Ju-Yeon Jo, "Phish Phactors: Offensive and Defensive Strategies". School of Informatics and Internet Forensics Laboratory University of Nevada, Las Vegas. Available at <http://www.berghel.net/publications/phishing/phishing.php> site last visited on 08 Jun 09.
- [2] Anti-Phishing Working Group, <http://www.antiphishing.org/index.html>
- [3] Ed Sperling, "Spear Phishing and Pharming", dt 21 Nov 08. Available at http://www.forbes.com/2008/11/21/phishing-pharming-cybertheft-identity08-tech-cx_es_1121phish.html, site last visited on 13 Jun 09.
- [4] Dhamija, R., Tygar, J.D. and Hearst, M., "Why Phishing Works" in Conference on Human factors in Computing Systems (SIGCHI 2006), pp. 581-590, 22-27 April 2006.
- [5] Chenfeng Vincent Zhou, Leckie C., Karunasekara S. and Tao Peng, "A Self-healing, Self-protecting Collaborative Intrusion Detection Architecture to Traceback Fast-flux Phishing Domains" in Networks Operations and Management Symposium (NOMS) Workshops 2008, Salvador, Brazil. IEEE, pp. 321-327, 7-11 Apr 2008.
- [6] Gartner Inc., "Gartner Survey Shows Phishing Attacks Escalated in 2007; More than \$3 Billion Lost to These Attacks", 2007 Press Release, 17-Dec-2007. Available at <https://www.gartner.com/it/page.jsp?id=565125>, site last visited on 08 Jun 09.
- [7] SearchSecurity.com Definitions, available at <http://searchsecurity.techtarget.com/dictionary/definition/1005812/attackvector.html> site last visited on 09 Jun 2009.
- [8] David Watson, Thorsten Holz and Sven Mueller "Know Your Enemy: Phishing", whitepaper published by 'The Honeynet Project', dated 16 May 2005. Available at <http://www.honeynet.org/papers/phishing/> site last visited on 09 Jun 2009.
- [9] Ollmann, G., "The Phishing Guide", 2005, NGS Software Insight Security Research. Available at <http://www.ngssoftware.com/papers/NISR-WP-Phishing.pdf>, site last visited on 09 Jun 09.
- [10] Hal Berghel, James Carpinter and Ju-Yeon Jo, "Phish Phactors: Offensive and Defensive Strategies". School of Informatics and Internet Forensics Laboratory, University of

Nevada, Las Vegas. Available at <http://www.berghel.net/publications/phishing/phishing.php>, site last visited on 09 Jun 09.

[11] http://searchsecurity.techtarget.com/sDefinition/0,,sid14_gci1134829,00.html site last visited 08 Jun 09.

[12] http://searchsecurity.techtarget.com/sDefinition/0,,sid14_gci1097059,00.html site last visited 08 Jun 09.

[13] <http://en.wikipedia.org/wiki/Vishing> site last visited on 08 Jun 09.

[14] “In Session Phishing Attacks” Trusteer Research paper dated 29 Dec 2008. Available at <http://www.trusteer.com/files/In-session-phishing-advisory-2.pdf> site last visited on 09 Jun 09.

[15] Bryan Parno, Cynthia Kuo and Adrian Perrig, “Phoolproof Phishing Prevention”, 03 December 2005. CY Lab, Carnegie Mellon University, Pittsburgh. Available at <http://www.cylab.cmu.edu/files/cmucylab05003.pdf>, site last visited on 09 Jun 09.

[16] Rachna Dhamija, J.D.Tygar, “The Battle Against Phishing: Dynamic Security Skins” in Proceedings of the symposium on Usable Privacy and Security, pp. 77-88, yr 2005.

[17] Engin Kirda and Christopher Kruegel, “Protecting Users Against Phishing Attacks” in Computer Software and Applications Conference, 2005 (COMPSAC 2005), Edinburgh, Scotland. 29th Annual International Volume 1, pp. 517 – 524, Issue: 26-28 Jul 2005.

[18] Juan Chen and Chuanxiong Guo, “Online Detection and Prevention of Phishing Attacks” in Communications and Networking in China, 2006 (ChinaCom '06) pp. 1-7, 25 -27 Oct 2006.

[19] Maher Aburrous, Hossain M.A., Thabatah F. and Dahal K., “Intelligent Phishing Website Detection System using Fuzzy Techniques” in Information and Communication Technologies: From Theory to Applications, 2008 (ICTTA 2008), 3rd International Conference. pp. 1-6, 7-11 Apr 2008.

[20] Chandrasekaran M., Ramkumar Chinchani and Shambhu Upadhyaya, “PHONEY: Mimicking User Response to Detect Phishing Attacks”, Proc., 2006 Int. Symposium on World of Wireless, Mobile and Multimedia Networks, pp. 668-672, 26-29 Jun 2006.

- [21] Yue Wang, Agrawal R. and Baek-Young Choi, "Light Weight Anti-Phishing with User Whitelisting in a Web Browser" in Region 5 Conference, 2008 IEEE, pp. 1-4, 17-20 Apr 2008.
- [22] <http://www.netbeans.org/community/releases/60/> site last visited 10 Jun 09.
- [23] <http://monkey.org/~jose/wiki/doku.php?id=PhishingCorpus> site last visited 10 Jun 09.
- [24] Scamdex: The Email Scam Resource available at <http://www.scamdex.com/> site last visited on 010 Jun 09.
- [25] Millersmiles: The Web's Dedicated Anti Phishing Service available at <http://www.millersmiles.co.uk>, site last visited on 10 Jun 09.
- [26] J. Zdziarski, W. Yang, and P. Judge, "Approaches to Phishing Identification Using Match and Probabilistic Digital Fingerprinting Techniques," Spam Conference 2006. Available at http://www.trustedsource.org/download/research_publications/phishing.pdf, site last visited on 11 Jun 09.

Publications

Kapil Oberoi and AK Sarje, "An Anti-Phishing Application for the End User", in proceedings of Hack.in, 3rd Hackers' Workshop at IIT Kanpur, pp. 17-23, 17- 19 Mar 2009.