Phishing E-mail Analysis

Shamal Firake, Pravin Soni, Dr. B.B.Meshram

shamal@inbox.com pravindsoni@gmail.com bmeshram@vjti.org.in

Abstract— The act of sending a forged e-mail (using a bulk mailer) to a recipient, pretending to be a legitimate in order to scam the recipient into divulging private information such as credit card numbers or bank account passwords is known as phishing. Seeking sensitive user data 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. According to Gartner estimates, 3.3% of the 124 million consumers who received phishing email last year were victimized and lost money because of the phishing email attacks. This paper is centered around Phishing Attacks on E-mail. Paper contains the brief literature review about the different approaches developed to detect and prevent phishing attacks. Paper gives basics of e-mail phishing attack, as how to send forged e-mails and how one can send mass emails to someone. We have also given the method for e-mail traversing which will be used for e-mail forensic analysis of forged e-mails. We have also given our proposed system for Detection and Prevention of Phishing Attacks on E-mail.

Keywords— *E-mail , Phishing Attack, E-mail Forging, Mass E-mailing, HyperLink Detection , Digital Signature.*

I. Introduction

Phishing has actually been around for over 15 years, starting with America Online (AOL) back in 1995.There were programs (like AOHell) that automated the process of phishing for accounts and credit card information.[2] Actually the term *phishing* is derived from the fact that Internet scammers "fish" for users' financial information and password data. "Ph" is a common replacement for the letter "f" in hacker lingo; one of the earliest forms of hacking was known as "phone phreaking."Phishing, in computer security field is described as the criminally fraudulent process that attempts to acquire sensitive information such as usernames, passwords and credit card details by masquerading as a trustworthy entity in an electronic communication. The frequently used attack method is to send e-mails to potential victims, which seemed to be sent by banks, online organizations, or ISPs. In these e-mails, they will makeup some causes, e.g. the password of your credit card had been mis-entered for many times, or they are providing upgrading services, to allure you visit their Web site to conform or modify your account number and password through the hyperlink provided in the e-mail. You will then be linked to a counterfeited Web site after clicking those links.

The phishing problem is a hard problem for a number of reasons. Most difficulties stem from the fact that it is very easy for an attacker to send mass emails or to do email forging to spoof e-mail addresses. Previous work indicates that the ability to create exactly similar looking copies of legitimate e-mails, as well as users' unfamiliarity with browser security indicators, leads to a significant percentage of users being unable to recognize a phishing attack. Unfortunately, the ease with which copies can be made in the digital world also makes it difficult for computers to recognize phishing attacks. As the phishing websites and phishing emails are often nearly identical to legitimate websites and emails, current filters have limited success in detecting these attacks, leaving users vulnerable to a growing threat.

According to the statistics provided by the Anti-Phishing Working Group (APWG) [1], in March 2010, email reports received by APWG from consumers were 30,577.The number of unique phishing sites detected ,in March 2010 were 29879.Payment Services returned to being the most targeted industry sector after Financial Services held top position during H2 2009. However, the category of ‚Other' rose from 13 percent to nearly 18 percent from Q4 2009 to Q1 2010, an increase of nearly 38 percent. Amongst the affected industry sector Payment services hold 37% and Financial services 35.9%[20].

In this paper we study the phishing e-mail analysis. We will see the basic steps how e-mail phishing attack is carried out. The easiest method to disguise the source of an e-mail and send it to the victim pretending to be a legitimate one is , E-mail Forging. Most attackers use this technique to fool the victim into believing that somebody else has sent the particular e-mail [34]. The basic experimentation of e-mail forging is also discussed. Another technique ,which is carried out e-mail phishing attack is Mass E-mailing. Now a days number of free softwares are there like MassMailer, BulkMail , eMailer, Mail me 3.00 etc. The only problem with these tools is that , they require to be on client side and require attacker to be stay online. To overcome this problem the Romanian Phishers found a lazy bulk mailing tool written in PHP [2]. This PHP bulk-mailing tool that executes on the server side, which utilizes the bandwidth of the compromised dedicated server. The PHP scripts used and their experimentation is included here. Thus by using these all techniques it is very easy to carry out e-mail phishing attacks that are the major threat in today's world. To overcome this problem till now many solutions are proposed which are really useful in their own sense. The brief discussion of all these approaches is done in literature survey .We also propose our own model to detect as well as prevent e-mail phishing attacks.

The rest of the paper is organised as follows. In section II the brief literature survey is provided which includes E-mail basics and approaches used to detect or prevent phishing attacks. In e-mail basics we have discussed e-mail header, mail delivery process and anonymous e-mails. Section III gives experimentation and analysis of email phishing attack. In section IV the proposed model to detect and prevent phishing attack is given. Section V concludes the paper.

II. LITERATURE SURVEY

A. *E-mail Basics*

E-mail contains specific key elements that enable it to communicate and route to the correct places. The design of the e-mail system is what makes e-mail one of the most efficient forms of communication today. Ironically, the e-mail system's infrastructure is similar to that of the traditional post office in that it requires you to have "routable" addresses enabling mail to be delivered. The mail server is similar to your human mail carrier, and the mail client is you physically walking to your mailbox.

1]. E-mail Header

Each time an email is sent on the Internet, it not only carries the message body , but also transmits relevant information on the path travel by it.This information is known as Email Header of the email. The most effective and easiest way to trace an email is to analyze its email headers. Most cybercrime investigators turn to email headers for evidence in any kind of e-mail related crime. All email communications on the internet is governed by rules and regulations laid down by two different protocols:

- Simple Mail Transfer Protocol(SMTP port 25)
- Post Office Protocol(POP port 110).

Each email on the internet originates at the sender's post office server with the help of SMTP commands.It is routed via number of intermediate mail servers and then finally reaches to the destination post office where the receiver use POP commands to download it to local system.Email headers are automatically generated and embedded into an email message both during composition and transfer between systems. They not only contain valuable information on the source of the email , but also represent the exact path taken by it, which can be represented as

Sender Outbox Source ⟶ Mail Server ⟶ Intermediate Mail Server ⟶ Destination Mail Server

⟶ **Destination Inbox**.

A typical email header looks something like this:

```
Return Path:
pankaj.chandigarh@gmail.com
Received: from
pankaj.chandigarh@gmail.com by
(208.50.6.127:25) via
   ug-out-1314.google.com
(66.249.92.171:20204) with [InBox.Com
SMTP Server]id 608240002387.WM27 for
shamal@inbox.com; Thu, 24 Aug 2006
00:34:37-0800
Received: by ug-out-1314.google.com
with SMTP id y2so389480uge        for
<shamal@inbox.com>; Thu, 24 Aug 2006
01:34:17 -0700 (PDT)
DomainKey-Signature: a=rsa-sha1; q=dns;
c=nofws;s=beta;d=gmail.com;
h=received:message-
id:date:from:to:subject:mime-
version:content-type;
b=Gq4HzLHTQwXmfvsJcJ65quwYgG9l/a6zLWwPB
r63PZ1WJE/8thjWVm+BD8GWCCg6Iu6/CdHYGggV
pcpXqmfNO4JDtVulPMkirNemUaSltKzjfuHF2DI
Ji1Zorhvq5CvxT10gTl92UjmQE5XMZkpHqGUBSm
X6O7Qwd27kfpZHjXo=
Received: by 10.67.119.13 with SMTP id
w13mr766642ugm;Thu, 24 Aug 2006
01:34:14 -0700 (PDT)
Received: by 10.66.237.1 with HTTP;
Thu, 24 Aug 2006 01:34:12 -0700 (PDT)
  Message-ID:
<1c058b0f0608240134s60d85438m67d196073f
8e1f14@mail.gmail.com>
  Date: Thu, 24 Aug 2006 14:04:12 +0530
  From: "Pankaj Mishra"
<pankaj.chandigarh@gmail.com>
  To: shamal@inbox.com,
shamal.firake@gmail.com
  Subject: Bjarne Stroustrup Book (C++)
  MIME-Version: 1.0
  Content-Type:multipart/mixed;
        boundary="----
=_Part_156068_33357618.1156408452495"
  X-Spam-Ratio: 0.02------
=_Part_156068_33357618.1156408452495
  Content-Type: multipart/alternative;
        boundary="----
=_Part_156069_14038231.1156408452495"

  ------
```

One can possibly identify the source of the email by simply Reverse Engineering the path travelled by it , which is explained in the next section.

2]. Mail Delivery Process

All e-mail headers contain the server and client information that controls the process of mail delivery. Many people who use e-mail clients have probably heard of SMTP servers and POP3 servers. Within the typical setup for e-mail, two ports are typically used: port 25, and port 110. Port 25 is the Simple Mail Transfer Protocol (SMTP), and its job is to transmit and receive mail—basically what is called a Mail Transfer Agent, or MTA.

An MTA is comparable to the mail carrier who picks up the mail and sends it off to where it needs to go. Just as the mail carrier drops off and picks up mail, so does the MTA. Port 110 is the Post Office Protocol, version 3 (POP3), and it is essentially the mailbox from which users pick up their mail up. The mail server infrastructure works in such an efficient fashion that we did not use only four servers but, at minimum, eight servers to deliver our e-mail. In the process of sending e-mail, we query multiple DNS servers to obtain information about where the mail servers are on the Internet. Here is an example of the complete process for sending an e-mail

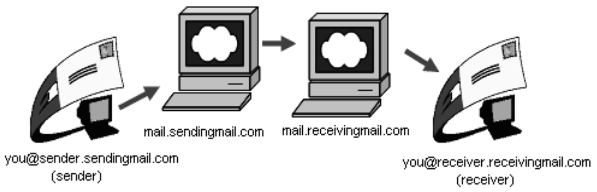


Figure 1 Standard Email Process

3]. Anonymous E-mail

Technology sector experts well know that SMTP was not designed with security in mind. E-mail is trivial to forge, and in more than one way, forged e-mail can be passed with ease to the mail transport agent (SMTP server). As we already are aware, spammers forge e-mails, and since phishers are classified as spammers, they take on this practice as well. Most spammers tend to forge e-mails for anonymity, since they are sending you annoying e-mails that will usually get a negative reaction, and if the e-mails were easily traceable, they would probably be caught.

Phishers forge for a different reason:They are attempting to con you, and they are using forgery to spoof a likely bank e-mail, such as verify@citibank.com. Not all headers can be forged, so the good news is that you can still track down the originator IP address, but unfortunately the phishers are not e-mailing directly from their homes.

The headers that can be forged are:

■ *Subject, Date, Message-ID*

■ Recipients: *From,To, CC*

■ Content body

■ Any arbitrary headers such as the *X-Mailer* and *X-Message-Info*

■ The initial *Received* headers

The headers that cannot be forged are:

■ The final *Received* headers

■ The originating mail server, including:

■ IP address

■ Subsequent timestamps

General clues within the header usually identify whether it is forged or not. The obvious one is the *Received* headers being inconsistent with mismatched *From* and *by* fields. The *HELO* name does not match the IP address, there are nonstandard headers in general placed within the e-mail, and wrong or “different” formats of the *Date, Received, Message-ID,* and other header labels.

B. Different Approaches Developed To Detect And Prevent Phishing Attacks

PILFER *et al.* [3] developed the tool *which* can be either deployed in a standalone configuration without a spam filter to catch a large percentage of phishing emails. PILFER Phishing email filter combines a set of features aimed at catching deception along with advanced machine learning techniques. The features used in approach includes Age of linked-to domains ,number of domains linked to, presence of attention-directing links (“click here”) that link to a domain other than the most common one in the email.

Engin Kirda et. al. **[6]** have developed an anti phishing solution 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 untrusted . 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 his 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. **[4]** have proposed an algorithm named LinkGuard which analyzes the generic characteristics of the hyperlinks in the phishing emails to deduce whether a site is spoofed or not. The algorithm makes use of a set of rules to analyze the URL viz, mismatch between the actual destination link and the link as seen by the user, use of IP addresses in dotted decimal format, absence of destination information in the text as seen by the user, etc.

Kapil et.el. **[16]** have developed an end user application that makes use of user provided data to check the authenticity of the destination URL and hence is able to give a more accurate prediction about the validity of the destination website. The approach save a mapping between supplied credentials and corresponding trusted website domains during the learning phase. In a detection phase, a submitted credential is matched with the saved credentials, and the current domain name is compared with the saved domain names. If there is a mismatch, a website is suspected as phishing.

Kristofer Beck and Justin Zhan [9] proposed Solution named Thin Client. A thin client is created to allow a secure connection between a client and the institution. We believe that this is a better way to prevent people from losing their private information due to phishing. A different interface than the traditional browsers which prove through past research are prone to fail in complete securityThis algorithm in itself may have faults because the phisher can theoretically take time and reengineer our thin client. This would be a change in the way phishers usually spoof a website by using phishing kits to replicate HTML code. It is harder to reengineer ActionScript used to create the thin client.

Ben Adida et.el.[13] suggested the Trusted Email Approach proposed the solution to authenticate certain email messages for the purpose of distinguishing legitimate business emails from spam and phishing attempts. All of the problems with spam and phishing start with SMTP. Due to its un-authenticated nature, anybody can send an email with a *From field equal to, for example,* ”billingsupport@companyXYZ.com”.A number of attempts have been made to add authentication to email. Most notably, PGP and S/MIME provide tools for encrypting and signing of email messages. A recipient of a signed message can verify the original sender based on the cryptographic signature. Unfortunately, neither PGP nor S/MIME would work on such large scale. The solution uses public key certificates for institutions only (though it does not require certificates) and does not require that users obtain certificates or public-private key pairs themselves .Unlike other solutions, *Trusted Email does not require* modifications to the Internet infrastructure (e.g. SMTP, DNS, etc.).

The proposed method of email verification is not designed to provide protection over already compromised communications channels. Lacking a trusted central key repository means that the initial communication between the user and the third party must be made without cryptographic verification of the third party’s identity. The *Trusted Email system is vulnerable* to a man-in-the-middle attack. It is also vulnerable to an eavesdropping attack where the attacker is able to eavesdrop the custom message, create his own email containing the attacker’s public key and the custom message and send the message so that it arrives before the original bank’s message.

Gansterer Polz et.el.[15] done e-mail classification for phishing defense based on different features of an-email. It is a classification-based approach for filtering phishing messages in an e-mail stream. Various features of every e-mail are extracted. This forms the basis of a classification process which detects potentially harmful phishing messages. The approach introduces new sixteen features. These newly introduced features belong to three different groups:

- The first group contains six “off-line” features.
- The second group contains eight “online” features.
- The third group is a control group of presumably class independent features containing two features: Subject length (SubjectLen) counts the number of characters in the subject field, and Sender length (SenderLen) counts the number of characters in the sender field of a message.

Chandrashekharan Krishnan et.el.[7] analyzed the structural properties of e-mail to separate phishing mails from legitimate e-mails. The main goal of approach is to classify phishing emails using a set of characteristics that remain relatively invariant across a large amount of emails. The characteristics used are language, layout, structure of phishing email so that all different contexts of phishing emails can be captured . The features relevant to language, composition and writing such as particular syntactic and structural layout traits, patterns of vocabulary usage, unusual language usage , stylistic and sub-stylistic features will remain relatively constant. Identifying and learning of these structural features with sufficiently high accuracy is very difficult challenge during phishing email classification.

III. Experimentation and Phishing Email Analysis

Reading e-mails has become a dangerous activity. E-mails can carry dangerous viruses, worms which can be executed by merely opening e-mail or clicking on active link or picture in an e-mail. This e-mail phishing attacks are easily carried out by email spoofing. The two techniques known as e-mail forging and mass emailing made the task very easy for phishers to grab more number of victims. In this section we will see the basic steps of e-mail phishing attack and the two mostly used above mentioned techniques to carry out these e-mail phishing attacks. The last part of section gives the method to trace the source of an e-mail and a way to detect and trace the forged e-mail.

A. Steps of E-mail Phishing Attack

• The attacker obtains E-mail addresses for the intended victims. These could be guessed or obtained from a variety of sources.

• The attacker generates an E-mail that appears legitimate and requests the recipient to perform some action.

• The attacker sends the E-mail to the intended victims in a way that appears legitimate and obscures the true source.

• Depending on the content of the E-mail, the recipient opens a malicious attachment, completes a form, or visits a web site.

• The attacker harvests the victim's sensitive information and may exploit it in the future.

Thus attacker obtains the e-mail addresses of victims from internet and address list that user believed to be private(CNET).To send these phishing e-mails to victims attackers may use E-mail Forging or Mass emailing which are very easy to implement.

B. *E-mail Forging*

E-mail forging allows an attacker to disguise the source of an e-mail and send it to the victim [34]. Attackers use this technique to fool the victim into believing that mail has came from some legitimate source. Unfortunately ,there is very little that a victim can do to counter e-mail forging other than remain cautious and alert.

The Simple Mail Transfer protocol (SMTP) is the de facto standard protocol used by e-mail clients and daemons to send e-mails on the Internet. This protocol is used by the SMTP daemon that by default runs on Port 25 of a mail server. Each time user writes an e-mail and clicks on the SEND button , the e-mail client automatically issues SMTP commands to the remote mail server and sends the specified message.

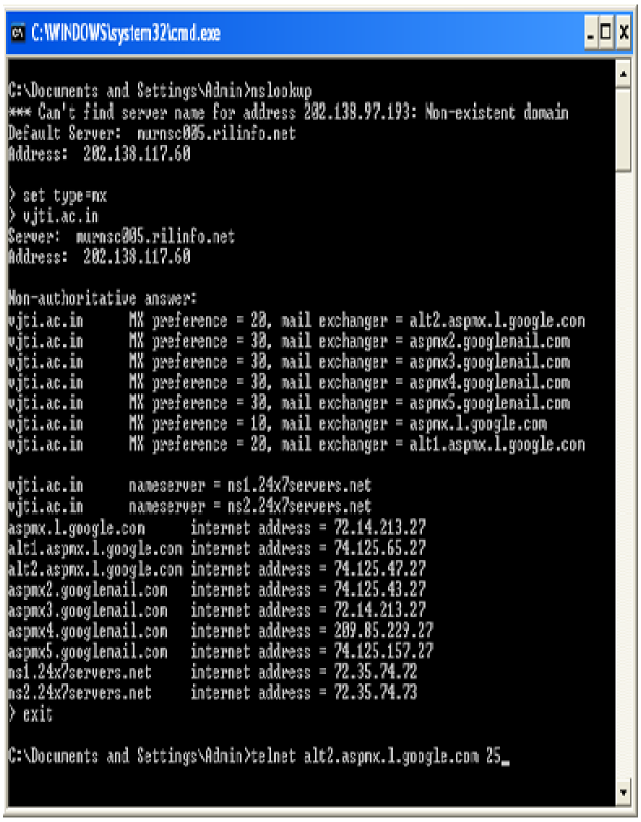
Unfortunately , the SMTP protocol also makes it extremely easy for an attacker to send forged e-mails to a remote user.It is quite possible for a user to connect manually to the SMTP port(25) of a remote mail server and use SMTP commands to manually send an artificial e-mail. This process of using SMTP commands to send e-mails from someone else's e-mail account is known as ***E-mail Forging.***

1]. The basic steps to carry out e-mail forging are:

- ***Step 1:*** **Open a command prompt**. In Windows, you can do this by clicking <Start>, <Run>, and type <cmd> in the box and press <OK>. You should get a black "Command Prompt" screen.
- ***Step 2:*** **You will need an SMTP server address to proceed**. Here is how to find one: On the command-line, type **<nslookup>**.
  - Then, type <set type=mx>.
  - Finally, enter the name of any website, for instance, <hazemdesigns.srhost.info>.
  - This will return the following: <Non-authoritative answer:hazemdesigns.srhost.info mail exchanger = 0 ASPMX.L.GOOGLE.COM.>.
  - The <aspmx.l.google.com> part is what you need. This is an SMTP server address.

  Type <exit> to exit out of nslookup. Fig 2 explains step 2.
- ***Step 3:*** Now after u have find out the smtp mail server type in cmd ***telnet "mail server" 25***

  In this case mail server is ***alt2.gmail-smtp-in.l.google.com***
- ***Step 4:*** Now we will be connected to the mail server, so now to begin, type in ehlo for esmtp (extended smtp) and helo for smtp type server in command window that appear after last step.



```
C:\Documents and Settings\Admin>nslookup
*** Can't find server name for address 202.138.97.193: Non-existent domain
Default Server:  murnsc005.rilinfo.net
Address:  202.138.117.60

> set type=mx
> vjti.ac.in
Server:  murnsc005.rilinfo.net
Address:  202.138.117.60

Non-authoritative answer:
vjti.ac.in      MX preference = 20, mail exchanger = alt2.aspmx.l.google.com
vjti.ac.in      MX preference = 30, mail exchanger = aspmx2.googlemail.com
vjti.ac.in      MX preference = 30, mail exchanger = aspmx3.googlemail.com
vjti.ac.in      MX preference = 30, mail exchanger = aspmx4.googlemail.com
vjti.ac.in      MX preference = 30, mail exchanger = aspmx5.googlemail.com
vjti.ac.in      MX preference = 10, mail exchanger = aspmx.l.google.com
vjti.ac.in      MX preference = 20, mail exchanger = alt1.aspmx.l.google.com

vjti.ac.in      nameserver = ns1.24x7servers.net
vjti.ac.in      nameserver = ns2.24x7servers.net
aspmx.l.google.com      internet address = 72.14.213.27
alt1.aspmx.l.google.com internet address = 74.125.65.27
alt2.aspmx.l.google.com internet address = 74.125.47.27
aspmx2.googlemail.com   internet address = 74.125.43.27
aspmx3.googlemail.com   internet address = 72.14.213.27
aspmx4.googlemail.com   internet address = 209.85.229.27
aspmx5.googlemail.com   internet address = 74.125.157.27
ns1.24x7servers.net     internet address = 72.35.74.72
ns2.24x7servers.net     internet address = 72.35.74.73
> exit

C:\Documents and Settings\Admin>telnet alt2.aspmx.l.google.com 25_
```

Figure 2. Using nslookup utility in Windows to get mail server address

- ***Step 5:*** Now here type exactly as given below :
  - Type:-helo mail server : the name of mail server
  - Type:- mail from: email id from where this mail is send from.eg. mail from:
  - Type:- rcpt to: email id to whom this mail go to.

    eg. rcpt to:
  - Now type:- data
  - Type the message that you want to send.
  - Now at last type .(dot) after entering data.

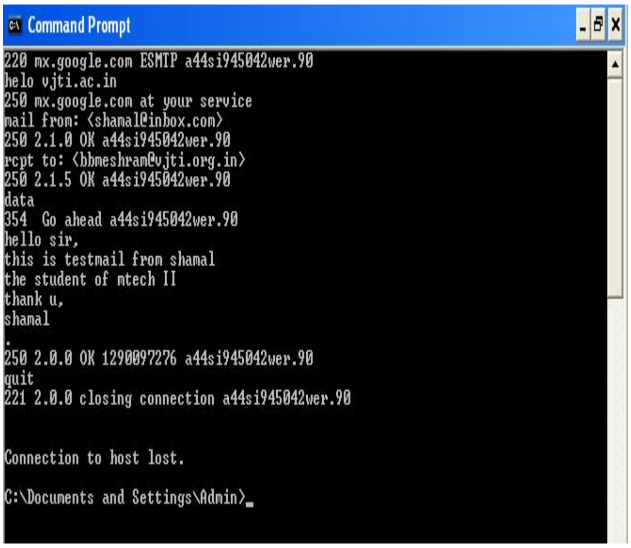
The step 4 and 5 are explained in fig 3.

In reality , one does not need to remember any SMTP commands. You can get help by simply typing HELP. To get the details of specific command type HELO followed by command name.

2]. Advanced E-mail forging

In the last section, we have seen the basic e-mail forging which is very easy to execute. However ,an attacker requires more control over the various features of the forged e-mail. Thus advanced features of e-mail forging includes:The Subject Field

- Sending File attachments
- The CC & BCC Fields
  - Using rcpt to Command
  - Using CC Field



```
220 mx.google.com ESMTP a44si945042wer.90
helo vjti.ac.in
250 mx.google.com at your service
mail from: <shamal@inbox.com>
250 2.1.0 OK a44si945042wer.90
rcpt to: <bbmeshram@vjti.org.in>
250 2.1.5 OK a44si945042wer.90
data
354  Go ahead a44si945042wer.90
hello sir,
this is testmail from shamal
the student of mtech II
thank u,
shamal
.
250 2.0.0 OK 1290097276 a44si945042wer.90
quit
221 2.0.0 closing connection a44si945042wer.90


Connection to host lost.

C:\Documents and Settings\Admin>_
```

Figure 3 Using SMTP commands to send fake e-mail

- **Using Subject Field**

Most professional and personal e-mails on the Internet have a suitable subject field describing the contents of the e-mail. Hence, from an attacker's perspective , in order to reduce suspicion , it is extremely important to send a forged e-mail with a subject .The SUBJECT argument is accepted by DATA command that is normally used to specify the content of forged e-mail. As soon as an attacker enters the DATA command the SMTP prompt is ready to accept both the contents of e-mail and also arguments if any. Fig 4 explains how to use subject field.

- **Sending File attachments**

All e-mail attachments were transmitted across networks using Unix-to-Unix encoding standard(UU-encoding standard).

UU-encoding Standard

- Transmit data safely without any corruption or loss of bytes.
- Converts data files into ASCII format
- Increases the size of any file by 42%

Thus files can be attached to forged e-mail by following the steps:

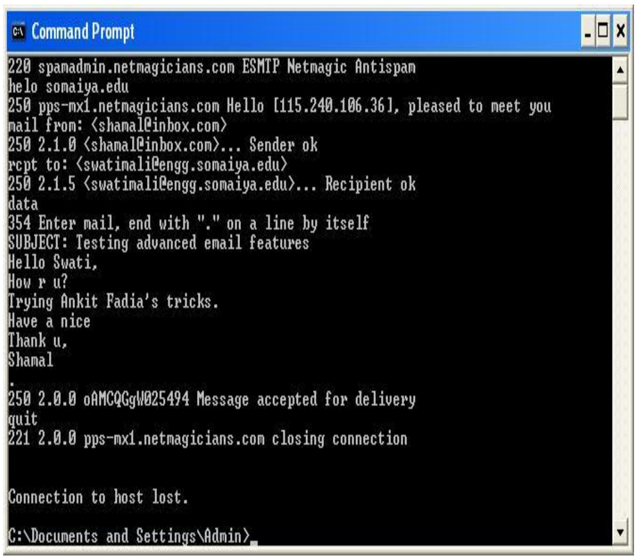
- Converting the file to be attached into the uuencoded format
- Connecting to the remote mail server and pasting the uuencode obtained in step1 into the DATA command.

Fig 5. explains how to send file attachments in e-mail forging.

- **Multiple Entries in TO Field**

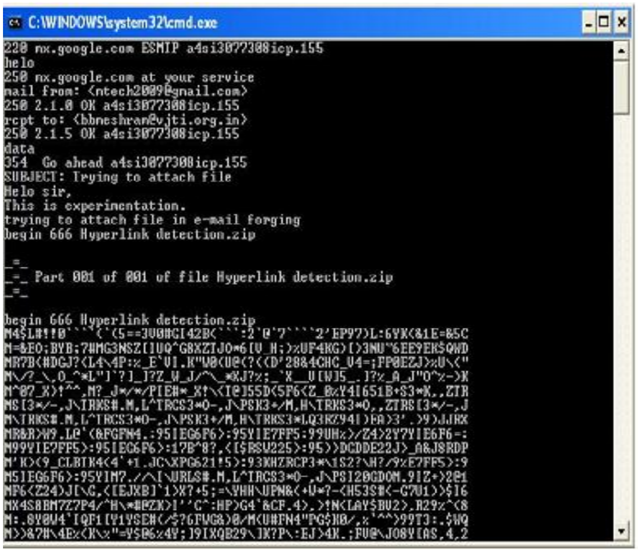
1. Connect and exchange introductions with mail server.
2. Use multiple RCPT commands to send the same e-mail to more than one persons.

The fig 6's telnet session demonstrates how to enter multiple e-mail addresses in the TO field



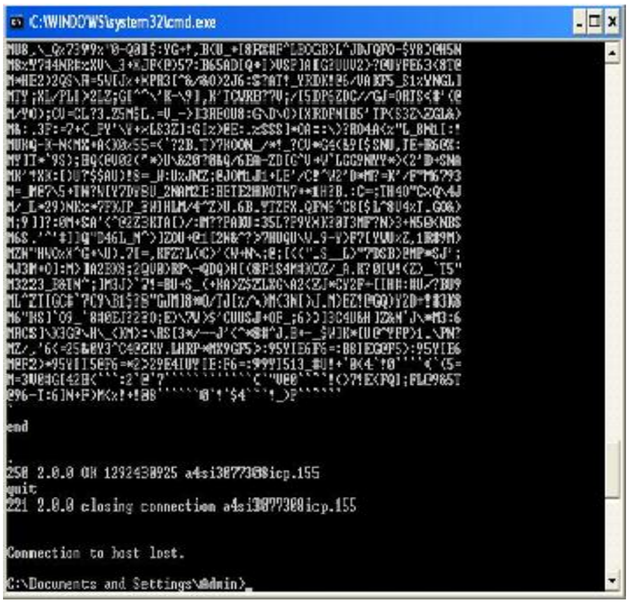
```
220 spamadmin.netmagicians.com ESMTP Netmagic Antispam
helo somaiya.edu
250 pps-mx1.netmagicians.com Hello [115.240.106.36], pleased to meet you
mail from: <shamal@inbox.com>
250 2.1.0 <shamal@inbox.com>... Sender ok
rcpt to: <swatimali@engg.somaiya.edu>
250 2.1.5 <swatimali@engg.somaiya.edu>... Recipient ok
data
354 Enter mail, end with "." on a line by itself
SUBJECT: Testing advanced email features
Hello Swati,
How r u?
Trying Ankit Fadia's tricks.
Have a nice
Thank u,
Shamal
.
250 2.0.0 oAMCQGgW025494 Message accepted for delivery
quit
221 2.0.0 pps-mx1.netmagicians.com closing connection


Connection to host lost.

C:\Documents and Settings\Admin>_
```

Figure 4 Advanced e-mail forging using subject field



```
220 mx.google.com ESMTP a4si3077308icp.155
helo
250 mx.google.com at your service
mail from: <ntech2009@gmail.com>
250 2.1.0 OK a4si3077308icp.155
rcpt to: <bbmeshram@vjti.org.in>
250 2.1.5 OK a4si3077308icp.155
data
354  Go ahead a4si3077308icp.155
SUBJECT: Trying to attach file
Helo sir,
This is experimentation.
trying to attach file in e-mail forging
begin 666 Hyperlink detection.zip

_=_
_=_ Part 001 of 001 of file Hyperlink detection.zip
_=_

begin 666 Hyperlink detection.zip
M4$L#!!0````(`(5==3V0#GI42B<```:2`@'7````2'EP97)L:6YK<&1E=&5C
M=&EO;BYB;7#MG3NSZ[IUQ^G8XZTJO*6[V_H;)%UF4KG)[)3NU"6EE9EK$QWD
MR7B<#DGJ?<L4\4P:%_E`V1.K"W0<U@<?<<D'28&4CHC_V4=;FP0EZJ>%U\<"
M\/?_\,0_^*L"]`?]_]?Z_W_J/^\_*KJ?%;_`X__U[W]5_.]?%_A_J"O^%->K
M^07_X)!^^,M?_J*/*/PIE#*_X!\<I@]55D<5F6<Z_0%Y4I651B+S3*K,,ZTR
MS[3*/-,J\IRKS#.M,L^TRCS3*O-,J\PSK3+/M,H\TRKS3*O,,ZTRS[3*/-,J
M\TRKS#.M,L^TRCS3*O-,J\PSK3+/M,H\TRKS3*LQ3KZ94I)EA>3'.>9)JJKK
MR&R>W9.L@'<&FGFW4.:95IEG6F6>:95YIE7FF5:99UH%>/Z4>2Y7YIE6F6=:
M99YIE7FF5>:95IEG6F6>:17B^8?,<[$RSV225>:95))DCDDE22J>_A&J8RDP
M'K)<9_CLBTK4<4`+1.JC\XPG621!5):93KHZRCP3*\1S2?\H?/9%E7FF5>:9
M5IEG6F6>:95YIM7./^I\URLS#.M,L^TRCS3*O-,J\PSI20GDOM.9IZ+)2@1
MF6<Z24)J[\G,<[EJXB]`1)X?+5;=\YHH\UPN&<+V*?-<H53S#<-G7U1)>$I6
MX4S8BM7Z7P4/^H\*#@ZK)I''C^:HP)G4`&CF.4>.>!N<LAY$BU2>.R29%^<8
M:.8Y0V4'IQF1[Y1YSEH</$?6FWG&)0/M<U#FN4"PG$H0/,%`^^>99T3:.$WQ
M>>&7#\4E%<K\%"=Y$@6%4Y;]9IXQB29\JK?P\:EJ>4K.;FU@\JO8Y[AS,4,2
```

Figure 5 Advanced e-mail forging to attach files(Part A)

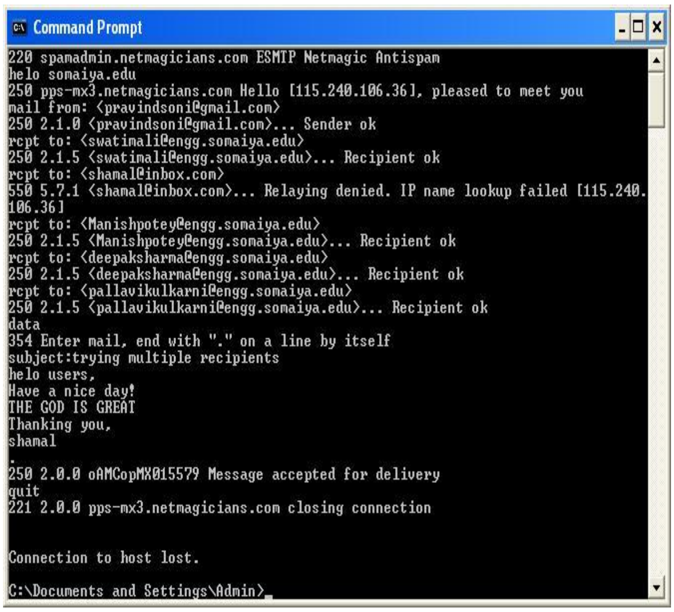


```
MU8,\_Q%73?9%'0-Q0I$:YG+!,B(U_+I8R&HF^LBOGB)L^JDJQPO-$Y8)C4H5N
M8%Y?44NR#%XU\_3+%JF(@)57:B65ADIQ+I)USP]AIG2UUV2)?@UYFE63<8T@
M*HE2)2QS\H=5V[J%+KPR3[^&/&0)2J6:S?AT!_YRDK!@6/VAKF5_S1%YNGLI
MTY;XL/PLI>2LZ;GI^^\'K-\9],K'TCVRB?7U;/I5DP6ZDC//GJ=ORTS<$'(@
M/Y0);CU=CL?3.Z5M$L.=U_->I3HE0U8:G\D\O)IXRDFNIB5'TP(S3Z\ZGL&)
M&:.3F:=7+C_PY'\Y+%LS3Z]:GI%)@E:.%SSS]*OA==\)?RO4A<%"L_BH1I:!
MUHQ-X-N(MX+A(X8%55=('?2B.T)?HOON_/*!_?CU*G4(&9[$SNU,TE+B6@%:
MY]I*'9S);BQ(@U82("*)U\&20?8&Q/6EA-ZD[G^U+V'LGG9NYY*)<2'D+SNA
MK'!XX:[)U?$$AU)!8=_H:U%JNZ;@JOM1J1+LE'/C@^V2'D*M?=K'/F"M6?93
M=_ME?\5+TN?V[Y7DYBU_2NAM2E:BETE2HNOTN?**1H?B.:C=;TH40"C%Q\4J
M/_L*29)NK%*7FXJP_2WJHLM/4^Z)U.6B.YTZFX.QFN6^CB[$L^8U4%T_GO&)
M;9]]?:0M+SA'<^@2Z3KJA[)/:M??PAXU=3SL?P9YXK?0T3MF?N)3+N5@(NBS
M6S.'^'#]]Q"D46L_M^>]ZOU+@1[2H&^?>?HUQU\V_9-Y)F7[YUU%Z,1R@9M)
MZH"HVO%X^G+\U).7I=,KFZ?L(C)'(W+N\;@;[<("_S__L)"7DSB)BMP*SJ';
MJ3M+O]:M>IA2EX8;2QVB)RP\-QDQ)H[(@F1S4M4XOZ/_A.K?0IV!(Z)_`T5"
M3223_B&IN^;]M3J)`?!=BU+S_(+HA)Z$ZLXG\A2<ZJ*CY2F-[IHH:HU/?BU9
ML^ZI[GC#`?C9\B1$?B"GJM]8*0/TJ[X/\)M(3N[)J.M)EZ!@GQ)Y2D+!I3]XS
M6"HS]`O9_'8H0EJ?2?0;E)\7V)S'CUUSJ+OF_;6>)]3C4U&H]Z&N'J\*M3:6
MACS]\X3G?\H\_(XM)=\RS[3*/--J'(^*@#^J.B+-_$VJK*[U@^YFP)1.\PN?
MZ/,'6<=25&@Y3^C4@ZKY.LHRP*MX9GF5>:95YIE6F6=:B8]EG@F5):95YIE6
M@F2)*95YII5@F6=*2>29E4IUYIE:F6=:99YI513_@U!+'0(4`!0````(`(5=
M=3V@4G[42H(```:2'@'7`````````````C`"V@@`````!()?!E(FQ];FLC9&5T
@96-I:6]N+F)M(%!+!BB``````@'!'$4`````!_)P``````
`
end

.
250 2.0.0 OK 1292430925 a4si3077308icp.155
quit
221 2.0.0 closing connection a4si3077308icp.155


Connection to host lost.

C:\Documents and Settings\Admin>_
```

Figure 6 Advanced e-mail forging to attach files(Part B)

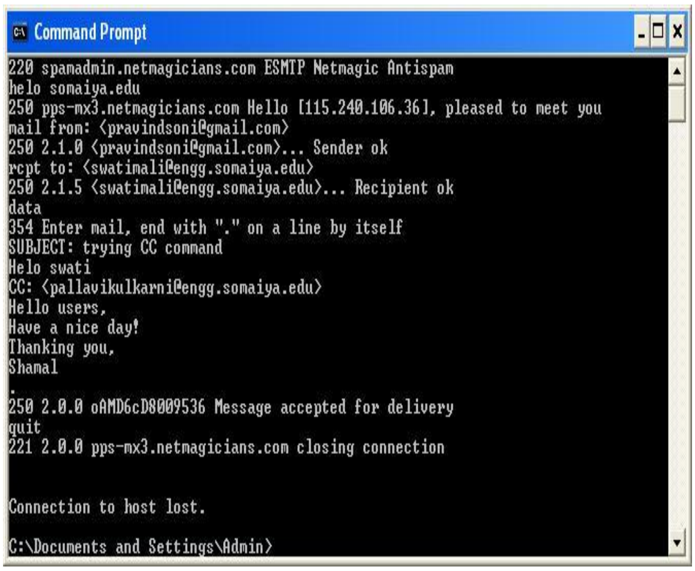
- Multiple Entries in TO Field and in CC Field

A user enters multiple e-mail addresses in both the TO field and the CC field , whenever wants to send the same e-mail to many people .The function of an entry in the CC field is equivalent to that of an entry in the TO field, even behind the scene SMTP working remains same. The e-mail addresses entered in the CC field are actually sent using multiple occurrences of the RCPT command. Example shown in fig 7 demonstrates how multiple entries can made in the CC field.



```
220 spamadmin.netmagicians.com ESMTP Netmagic Antispam
helo somaiya.edu
250 pps-mx3.netmagicians.com Hello [115.240.106.36], pleased to meet you
mail from: <pravindsoni@gmail.com>
250 2.1.0 <pravindsoni@gmail.com>... Sender ok
rcpt to: <swatimali@engg.somaiya.edu>
250 2.1.5 <swatimali@engg.somaiya.edu>... Recipient ok
rcpt to: <shamal@inbox.com>
550 5.7.1 <shamal@inbox.com>... Relaying denied. IP name lookup failed [115.240.
106.36]
rcpt to: <Manishpotey@engg.somaiya.edu>
250 2.1.5 <Manishpotey@engg.somaiya.edu>... Recipient ok
rcpt to: <deepaksharma@engg.somaiya.edu>
250 2.1.5 <deepaksharma@engg.somaiya.edu>... Recipient ok
rcpt to: <pallavikulkarni@engg.somaiya.edu>
250 2.1.5 <pallavikulkarni@engg.somaiya.edu>... Recipient ok
data
354 Enter mail, end with "." on a line by itself
subject:trying multiple recipients
helo users,
Have a nice day!
THE GOD IS GREAT
Thanking you,
shamal
.
250 2.0.0 oAMCopMX015579 Message accepted for delivery
quit
221 2.0.0 pps-mx3.netmagicians.com closing connection


Connection to host lost.

C:\Documents and Settings\Admin>_
```

Figure 7 Advanced e-mail forging to include multiple entries in TO field to send mail to multiple recipients.



```
220 spamadmin.netmagicians.com ESMTP Netmagic Antispam
helo somaiya.edu
250 pps-mx3.netmagicians.com Hello [115.240.106.36], pleased to meet you
mail from: <pravindsoni@gmail.com>
250 2.1.0 <pravindsoni@gmail.com>... Sender ok
rcpt to: <swatimali@engg.somaiya.edu>
250 2.1.5 <swatimali@engg.somaiya.edu>... Recipient ok
data
354 Enter mail, end with "." on a line by itself
SUBJECT: trying CC command
Helo swati
CC: <pallavikulkarni@engg.somaiya.edu>
Hello users,
Have a nice day!
Thanking you,
Shamal
.
250 2.0.0 oAMD6cD8009536 Message accepted for delivery
quit
221 2.0.0 pps-mx3.netmagicians.com closing connection


Connection to host lost.

C:\Documents and Settings\Admin>
```

Figure 8 Advanced e-mail forging to include CC field to send mail to multiple recipients

C. *Mass Emailing*

Phishers can use readymade bulk mailing tools available on net or they can build on their own. Now a days number of free softwares are there like MassMailer , BulkMail , eMailer, Mail me 3.00 etc. Most of the phishers found a lazy but efficient bulk-mailing method that does not require them to stay on the Internet while the bulk mailings are being sent. Most bulk-mailing tools are client side and require the client computer to be on the Internet while sending the e-mails. So Phishers use a PHP bulk-mailing tool that executes on the server side, which utilizes the bandwidth of the compromised dedicated server. This bulk mailing tool include four files as

1. Mail.php
2. Ini.inc
3. Maillist.txt
4. Testmail.html

Mail.php

```
<?php

include("ini.inc");

$mail_header = "From: mtechcomp2009@gmail.com";

$mail_header .= "Content-Type: text/html\n";

$subject="Account Verification Requested";

$body=loadini("testmail.html");

if (!($fp = fopen("maillist.txt", "r")))

exit("Unable to open mailing list.");

$i=0;

print "Start time is "; print date("Y:m:d H:i:s"); print
"\n";

while (!feof($fp)) {

fscanf($fp, "%s\n", $name);
```

```
print $name;
$i++;
mail($name, $subject, $body, $mail_header);
}
print "End time is "; print date("Y:m:d H:i:s");
?>
```

Ini.inc

The include file ini.inc, which is a header file that contains the functions we are calling within the bulkmail.php program.

```
<?php
function loadini($path) {
$fp = fopen($path, "r");
$fpcontents = fread($fp, filesize($path));
fclose($fp);
return $fpcontents;
}
function readini($filename, $key) {
return rfi($filename,$key,TRUE);
}
function rfi($filename, $key, $just_value) {
$filecontents=loadini($filename);
$key .= "=";
$currentkey = strstr($filecontents, $key);
if (!$currentkey)
return($empty);
$endpos = strpos($currentkey, "\r\n");
if (!$endpos) $endpos = strlen($currentkey);
if ($just_value) $currentkey = trim(substr($currentkey, strlen($key),
$endpos-strlen($key)));
else $currentkey = trim(substr($currentkey, 0, $endpos));
return ($currentkey);
}
?>
```

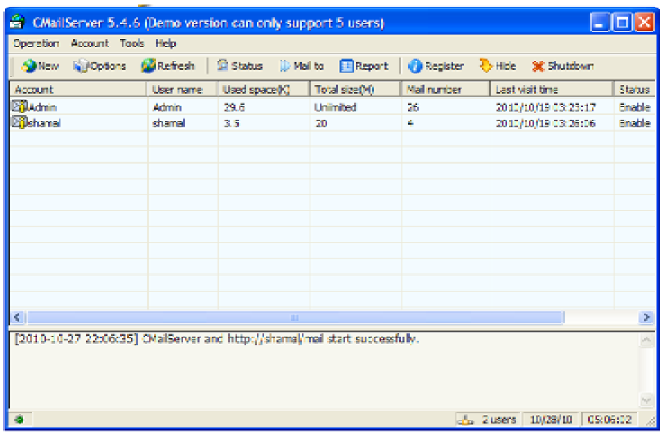
maillist.txt : The maillist.txt is a text file with the list of e-mail addresses that we plan to send to the victims.

admin@shamal.com
admin@shamal.com
admin@shamal.com
admin@shamal.com

testmail.html: The testmail.html is the e-mail we are sending.

The mail.php program has two ways of execution; via our Web browser or the command line. The command line will require us to be on the server shell and execute it, whereas with the Web browser, the phisher can hit it and exit the browser, leaving the server to do the rest of the work.

We have used CMailServer as our local mail server and run the above PHP scripts on Wamp server's localhost. Fig 8 shows that using above script we can successfully send mass emails to admin@shamal.com.


CMailServer 5.4.6 (Demo version can only support 5 users)
Operation Account Tools Help
New Options Refresh Status Mail to Report Register Hide Shutdown

Account	User name	Used space(K)	Total size(M)	Mail number	Last visit time	Status
Admin	Admin	29.6	Unlimited	26	2010/10/19 03:23:17	Enable
shamal	shamal	3.5	20	4	2010/10/19 03:26:06	Enable

[2010-10-27 22:06:35] CMailServer and http://shamal/mail start successfully.

2 users 10/28/10 05:06:22

Figure 9 Mass E-mailing using CmailServer

D. Method to trace the source of an e-mail.

Each time an email is sent on the Internet, it not only carries the message body , but also transmits relevant information on the path travel by it.This information is known as Email Header of the email.Email headers are automatically generated and embedded into an email message both during composition and transfer between systems. They not only contain valuable information on the source of the email , but also represent the exact path taken by it, which can be represented as

Sender Outbox → Source Mail Server → Intermediate Mail Server → Destination Mail Server → Destination Inbox.

One can possibly identify the source of the email by simply Reverse Engineering the path traveled by it.Most Cyber Crime investigators turn to email headers for evidence in any kind of email related crime. The above email header can be divided into the following chunks:

Part A:

```
Return Path:
pankaj.chandigarh@gmail.com
Received: from
pankaj.chandigarh@gmail.com by
(208.50.6.127:25) via
   ug-out-1314.google.com
(66.249.92.171:20204)             with
[InBox.Com      SMTP      Server]id
608240002387.WM27                  for
shamal@inbox.com; Thu, 24 Aug 2006
00:34:37-0800
Received: by ug-out-1314.google.com
with    SMTP    id    y2so389480uge
for <shamal@inbox.com>; Thu, 24 Aug
2006 01:34:17 -0700 (PDT)
DomainKey-Signature:     a=rsa-sha1;
q=dns; c=nofws;s=beta; d=gmail.com;
h=received:message-
id:date:from:to:subject:mime-
version:content-type;
b=Gq4HzLHTQwXmfvsJcJ65quwYgG9l/a6zLW
wPBr63PZ1WJE/8thjWVm+BD8GWCCg6Iu6/Cd
HYGggVpcpXqmfNO4JDtVulPMkirNemUaSltK
zjfuHF2DIJi1Zorhvq5CvxT10gTl92UjmQE5
XMZkpHqGUBSmX6O7Qwd27kfpZHjXo=
 Received:  by  10.67.119.13  with
SMTP  id  w13mr766642ugm;Thu, 24 Aug
2006 01:34:14 -0700 (PDT)
Received: by 10.66.237.1 with HTTP;
```

Part A:

Return-Path: pankaj.chandigarh@gmail.com

- This email address is the sender's email address.
- The Source Mail Server: ug-out-1314.google.com ([66.249.92.171])
- The Destination Mail Server : inbox.com
- The receiver connects to this destination mail server and download the email using simple POP command.
- Thus the complete path travelled by email can be depicted in the following manner
- (Source) 10.66.237.1-----→(Source Mail Server) ug-out-1314.google.com ([66.249.92.171]) ----→ (Destination Mail Server) inbox.com ---→TARGET SYSTEM(Destination).

Part B:

```
 Message-ID:     <20100727143839.30156.@
mail.gmail.com >Date:  Thu,  24  Aug
2006 14:04:12 +0530
 From:         "Pankaj        Mishra"
<pankaj.chandigarh@gmail.com>
 To:                shamal@inbox.com,
shamal.firake@gmail.com
 Subject:  Bjarne  Stroustrup  Book
(C++)
 MIME-Version: 1.0
 Content-Type: multipart/mixed;
 X-Spam-Ratio:            0.02------
=_Part_156068_33357618.115640845249
5
 Content-Type:
multipart/alternative;
        boundary="----
=_Part_156069_14038231.115640845249
5"

 Content-Transfer-Encoding: 7bit
```

Part B : It tells us that email was sent by pankaj.chandigarh@gmail.com

MessageId : MessageId field of the email header can be broken down in the following manner.

- 20100728080538: The email was sent in the year 2010, month July(7th),day 28th and at the time 14 hours, 38 minutes and 39 seconds.
- 30156: Each email send by the Mail Server has unique message Id reference number associated with it.The log file contain all the message Id's.

Cyber crime Investigators often use the reference number to carry out investigations.

Now up till now we have seen the basics of e-mail, details provided by e-mail header and methods used to send fake e-mails. On the basis of the above study we can give now the algorithm for e-mail tracing as follows:

1]. Algorithm For Email Tracing

- Step 1. Open the Email header.

// The SMTP protocol is used to send emails while the POP protocol is used to receive them.

- Step 2.Identify the source and destination of email by tracing the path

Sender Outbox ----→ Source Mail Server----→Intermediate Mail Server ---→Destination Mail Server ---→ Destination Inbox.

- Step 3.Identify the IP Address of the computer that was used to send the email with the help of Unique Message ID reference stored on the log file on a Mail Server. **OR**

Step 3 can be performed as below

// Use Reverse DNS look up ie convert the suspected IP Address into the corresponding hostname.

Use utility named *nslookup.*

$>nslookup IP Address of the sender

$>nslookup 203.94.243.71

203.94.243.71 has valid reverse DNS of mail2.mtnl.net.in

OR

Step 3 : Write a program to convert the IP Address to hostname or vice versa using JAVA coding IPAddress API

2]. Method to Trace forged E-mail

The above algorithm gives us the physical source of an e-mail , provided it is sent by an authorized user. To trace a fake e-mail sent by using e-mail forging or mass e-mailing we need more details again. To trace these e-mails following steps can be followed.

- Check the final received header field , because it can not be forged in any case. So if the DNS names are different in sender's e-mail address and final received header that means it is a forged e-mail.
- Now to trace it one can examine tcp_wrapper, ident, and sendmail logs to obtain information on the origin of the spoofed email.
- The header of the email message often contains a complete history of the "hops" the message has taken to reach its destination. Information in the headers (such as the "Received:" and "Message-ID" information), in conjunction with your mail delivery logs, should help you to determine how the email reached your system.
- If your mail reader does not allow you to review these headers, check the ASCII file that contains the original message.

This process may help you to trace a forged or fake e-mail but realize that in some cases, you may not be able to identify the origin of the spoofed email.

IV. Proposed Model To Detect And Prevent Phishing Attacks

Now a days phishing e-mail attacks are very easy for fradulents to carry out. As mass e-mails are sent , the number of affected victims are also large. To fight against such attacks, we proposed an anti-phishing tool to detect and prevent e-mail phishing attacks.

Problem Statement :

Detection and Prevention of Phishing Attacks on Email.

Fig. 9 Shows the basic architecture of the tool.

A. Modules of the Application

The tool mainly implements following modules

1. User Interface Module

User friendly graphical interface will be developed by using java technology for ease of use. It facilitates the use of tool for naive users also.

.

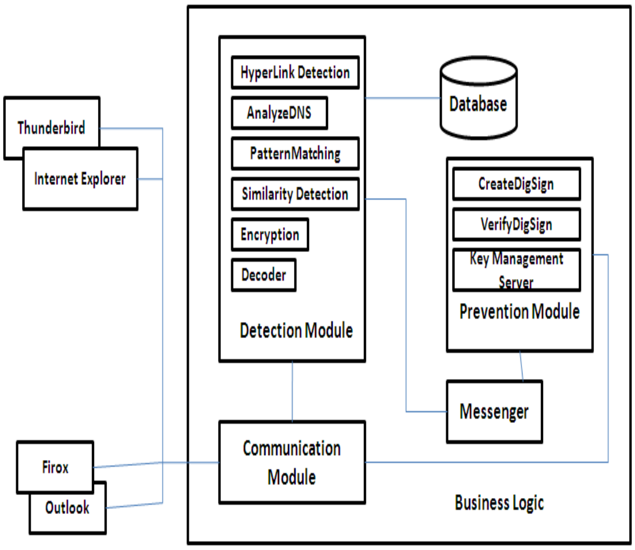


Figure 10 Arhitecture of the tool to detect and prevent phishing attacks

2. Database Maintenance Module

Data storage module storing, managing and if needed update the URL and IP address information of trusted websites

3. Business Logic Module

This module implements Hyperlink Detection Module. It uses data provided by user emails and database. It contains sub modules as

- Detection Module
- Prevention Module
- Communication Module
- Messenger Module

3.1 Detection Module

Detection Module reads the mails from inbox of mail client of user. It scans all messages and detects for any phishing attack , by using generic characteristics of

Hyperlinks. The Detection Module includes following sub-modules,

a. Hyperlink Detection Module

Hyperlink Detection Module fetches the DNS names of actual link and visual links of hyperlinks. If both the links are not empty and are different then it warns user about the phishing attack. Again it checks whether the actual DNS name is directly used as dotted decimal then returns possible phishing attack. Many times to confuse the user the actual links and visual links are encoded by using Hexadecimal code or ASCII code. To handle such situations module calls the respective DECODER modules and then compare the decoded links. Module also checks the JavaScript attack if any present in an email. It just checks for the keyword ”Java Script” in the email text and if it present, module warns user as possible phishing Attack. This module implements DetectHyperLink Algorithm.

b. AnalyzeDNS Names Module

AnalyzDNS Names module is used if visual link is null in the hyperlink. The module then check the DNS of hyperlink in Blacklist and whitelist respectively. If It doesn’t find there also, then it calls the pattern matching module.This module is implemented using AnalyzeDNS algorithm.

c. PatternMatching Module

It implements the PatternMatching algorithm. Pattern matching module first extracts the DNS name from sender’s email address.If this senders DNS name and actual link DNS name are different then it is possibly a phishing attack.If both are same then module checks the previously accessed links database maintained as SEED_SET .SEED_SET is a list of possible phishing links previously accessed or identified. Module then checks the DNS name of actual link against each item in the SEED_SET. If match is found module returns as POSSIBLE PHISHING attack .To compare it calls the Similarity algorithm module.

d. Similarity Detection Module

This module checks how much similar is the actual link DNS name with an item in the SEED_SET. If similarity is beyond a threshold value then it returns true otherwise false. This module uses Similarity algorithm.

e. Encryption Module

It Implements MD5 algorithm to calculate message digest of URL and IP addresses of such institutions/websites where he sends his login details, i.e., username and password. **MD5** (**Message-Digest algorithm 5**) is a widely used cryptographic hash function with a 128-bit hash value. MD5 is commonly used to check the integrity of files and has been employed in a wide variety of security applications.

f. DECODER Module

This module consist of two parts as

- **Hexadecimal Decoder**

Many times the hyperlinks are mentioned in Hexadecimal format so that normal user may get confused. To understand the actual DNS name of encoded hyperlink , it must be first decoded. Hexadecimal Decoder algorithm decodes the given Hexadecimal given format into normal text.

- **ASCI I Decoder**

Likewise , the hyperlinks are mentioned in Hexadecimal format also so that normal user may get confused. To understand the actual DNS name of encoded hyperlink , it must be first decoded. The algorithm decodes the given ASCII given format into normal text.

3.2 Prevention Module

Prevention Module helps user to prevent from phishing attacks. It allows user to create Digital Signatures to send the official messages .The receiver will verify the Digital Signature at other end and authenticates the sender of the message. A message signature is essentially a sophisticated one-way hash value that uses aspects of the sender’s private key, message length, date and time. In general the module does the following things

- Create a personal public/private key pair

Upload their public key to respected key management servers so that other people who may receive emails from the user can verify the messages integrity.

- Enable, the automatic signing of emails

Verify all signatures on received emails and be careful of unsigned or invalid signed messages – ideally verifying the true source of the email

3.3 Communication Module

Communicate with all of the monitored processes, collect data related to user input from other processes (e.g. IE, outlook, firefox, etc.), and send these data to the Business Logic module, it can also send commands (such as block the phishing sites) from the Business Logic executive to other modules.

3.4 Messenger

When receiving a warning messages from Business Logic module , it shows the related information to alert the users and send back the reactions of the user back to the Business Logic module.

V. CONCLUSION

The specter of online identity threat was never so real as it is today primarily due to rapid growth of the Internet and increase in online trading activities which offer a cost effective method to service providers, such as banks, retailers etc., to reach out to their customers via this medium. This has also provided the phishing community an excellent tool to try and fool the netizans into divulging sensitive information about their banking accounts, credit cards details, etc. Recent years have witnessed a host of phishing scams with each doing the other in terms of reach to the users and the level of sophistication.

Though the best measure available against such scams is user awareness , it is highly impossible also. So many tools have been developed to fight against the e-mail phishing attacks. To contribute in this regard we, have also taken a step ahead. This paper gives the details literature survey of the approaches till now used by different people to detect and prevent e-mail phishing attacks. We have given the details of how e-mail phishing attacks are carried out with experimentation results. We have also proposed our own approach to fight against the e-mail phishing attacks. The modules include the detail specification of their functionality. Thus , we assure that this solution will help to the normal end user as well as the corporate people also to send the highly confidential data.

REFERENCES

1. The Anti-phishing working group. http://www.antiphishing.org/.
2. A. Williams. "Phishing Exposed". Syngress Publishing Inc.; 2005.
3. I. Fette, N. Sadeh, and A. Tomasic. "Learning to detect phishing emails". Technical Report CMU-ISRI-06-112, Institute for Software Research, Carnegie Mellon University, June 2006. http://reports-archive.adm. cs.cmu.edu/anon/isri2006/abstracts/06-112.html
4. Juan Chen and Chuanziong Guo "Online Detection and Prevention of Phishing Attacks" IEEE Communications and Networking, ChinaCom '06, pp 1-7, Oct 2006.
5. Ollmann, G., "The Phishing Guide". NGS Software Insight Security Research 2005, http://www.ngs software.com/papers/NISRWPhishing.pdf.
6. 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 July 2005.
7. M. Chandrashekaran, K. Narayana, S. Upadhyaya,"Phishing Email Detection Based on Structural Properties", s*ymposium on Information Assurance: Intrusion Detection and Prevention*,New York, 2006
8. R. Suriya1 , K. Saravanan2 and Arunkumar Thangavelu An Integrated Approach to Detect Phishing Mail Attacks A Case Study, 3, *SIN'09,* October 6–10, 2009, North Cyprus, Turkey. Copyright 2009 ACM 978-1-60558-412-6/09/10
9. Phishing in Finance, Kristofer Beck, Justin Zhan, 978-1-4244-6949-9/10/$26.00 ©2010 IEEE
10. Huajun Huang, Shaohong Zhong, Junshan Tan, Browser-side Countermeasures for Deceptive Phishing Attack, 2009 Fifth International Conference on Information Assurance and Security
11. Danesh Irani, Steve Webb, Jonathon Giffin and Calton Pu," Evolutionary Study of Phishing" 978-1-4244-2969-1/08/ c_ 2008 IEEE
12. Jordan Crain, Lukasz Opyrchal, Atul Prakash,"Fighting Phishing with Trusted Email"**,** 2010 International Conference on Availability, Reliability and Security
13. B. Adida, S. Hohenberger, and R. L. Rivest, "Fighting phishing attacks: a lightweight trust architecture for detecting spoofed emails," February 2005, draft.
14. P. Kumaraguru, Y. Rhee, A. Acquisti, L. F. Cranor, J. Hong, and E. Nunge. Protecting people from phishing: the design and evaluation of an embedded training email system. In CHI '07: Proceedings of the SIGCHI conference on Human factors in computing systems, pages 905–914, New York, NY, USA, 2007. ACM.
15. Wilfried N. Gansterer, David P¨olz," E-Mail Classification for Phishing Defense".
16. Kapil Oberoi and Anil K. Sarje , "An Anti-Phishing Application for the End User" 3rd Hackers' Workshop on Computer and Internet Security March 17-19, 2009, Prabhu Goel Research Centre for Computer & Internet Security Department of Computer Science and Engineering Indian Institute of Technology Kanpur
17. Weider D. Yu Shruti Nargundkar Nagapriya Tiruthani," **PhishCatch – A Phishing Detection Tool",** 2009 33rd Annual IEEE International Computer Software and Applications Conference.
18. Phishing in Finance, Kristofer Beck, Justin Zhan, 978-1-4244-6949-9/10/©2010 IEEE
19. Phishing Activity Trends Report, 2009, Available online: http://www.antiphishing.org/reports/apwg_report_ 2009.pdf
20. Phishing Activity Trends Report, 1st Half 2010, Available online: http://www.antiphishing.org/reports/apwg_report_h1_2010.pdf
21. Mohamad Badra, Samer El-Sawda, Ibrahim Hajjeh," Phishing Attacks and Solutions**"** *Mobimedia'07*, Month 8, 2007, Nafpaktos, Aitolokarnania, Greece. Copyright 2007 ICST 978-963-06-2670-5
22. *Soroush Dalili,"* How to prevent phishing attacks?- In 3 Pages -
23. Sun Bin, Wen Qiaoyan, Liang Xiaoying," A DNS based Anti-Phishing Approach", 2010 Second International Conference on Networks Security, Wireless Communications and Trusted Computing
24. David Harley,"A Preety Kettle Of Phish".
25. "Today's Blended Threats", White Paper, Symantec Internet Security Threat Report, Trends for July-December 06, p. 13.
26. Tod Beardsley,"Phishng Detection and Prevention ,A practical Counter Fraud Solutions".
27. "Discover What Your Boss Is Looking At"
28. Lightweight Signatures for Email By Ben Adida_ David, Chau_ Susan ,Hohenberger_,† Ronald L. Rivest, Computer Science and Artifical Intelligence Laboratory.
29. Phishing Secrets: History, E®ects, and Countermeasures, Antonio San Martino and Xavier Perramon, *International Journal of Network Security, Vol.11, No.3, PP.163{171, Nov. 2010*

30. Putting an End to Account-Hijacking Identity Theft Federal Deposit Insurance Corporation ,Division of Supervision and Consumer Protection ,Technology Supervision Branch ,December 14, 2004

31. Phishing attacks and Countermeasures, Anil Sagar,Operations Manager

32. Anti-Phishing *Best Practices for Institutions and Consumer,Mccafe*

33. A Practical Approach to Managing Phishing Michael Barrett, Chief Information Security Officer Dan Levy, Senior Director of Risk Management – EuropeApril 2008

34. “E-MAIL HACKING” By Ankit Fadia.