International Journal of Applied Engineering and Technology ISSN: 2277-212X (Online)
An Open Access, Online International Journal Available at http://www.cibtech.org/jet.htm
2015 Vol. 5 (1) January-March, pp.59-68/Singh and Kaur
Research Article

IMPLEMENTATION FOR TRACING OF EMAIL

***Gurpreet Singh[1] and Manupreet Kaur[2]**
Department of Computer Science Engineering
**Author for Correspondence*

ABSTRACT
Email is an information and communications technology. It uses technology to communicate a digital message over the Internet. Users use email differently, based on how they think about it. There are many software platforms available to send and receive. Popular email platforms include Gmail, Hotmail, Yahoo! Mail, Outlook, and many others. The proposed work is to develop an algorithm to that works with all modern email companies including Hotmail, Gmail, Yahoo, AOL etc and all client side email programs including Outlook, Eudora etc. We do not need to download any software or plug-in to use our Email tracking Portal, just send the emails in the same way as we send now and find the results that how many recipient opened the email and how many times a single recipient opened it.

Keywords: *Email, Internet, Tracking, HTTP, SMTP*

INTRODUCTION
Email, short for Electronic Mail, consists of messages which are sent and received using the Internet. E-mail system comprises of various hardware and software components that include sender's client and server computers and receiver's client and server computers with required software and services installed on each. Besides these, it uses various systems and services of the Internet. The sending and receiving servers are always connected to the Internet but the sender's and receiver's client connects to the Internet as and when required.
An e-mail communication between a sender 'Alice' having e-mail address 'alice@a.com' and recipient 'Bob' having e-mail address 'bob@b.com' is shown in Fig.1

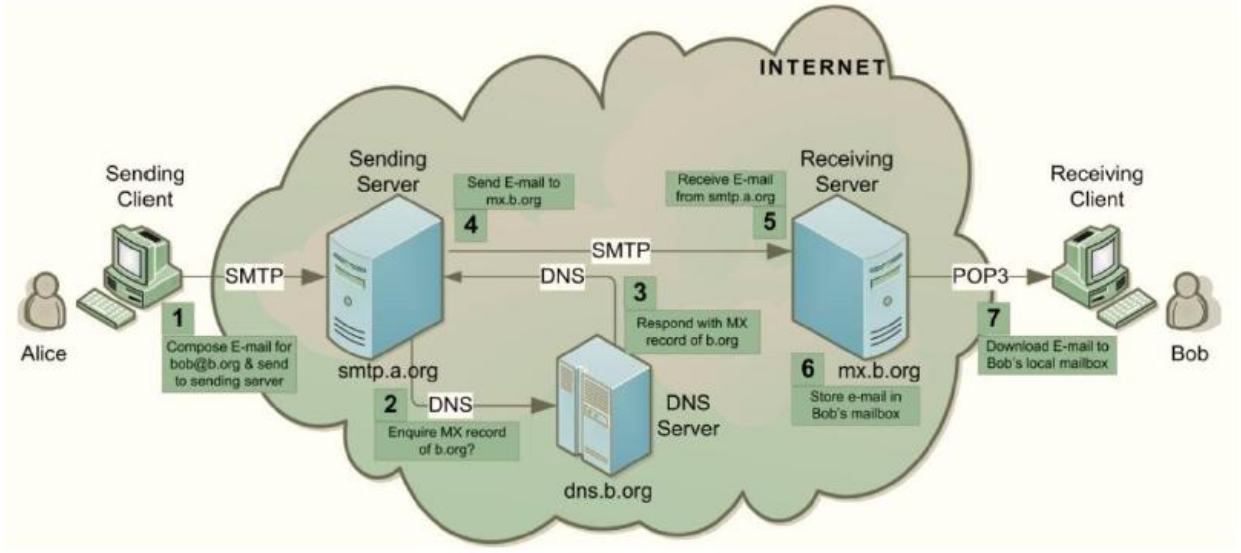

Figure 1: E-mail communication between a sender 'Alice' and recipient 'Bob'
'Alice' composes an e-mail message on her computer called client for 'Bob' and sends it to her sending server 'smtp.a.org' using SMTP protocol. Sending server performs a lookup for the mail exchange record of receiving server 'b.org' through Domain Name System (DNS) protocol on DNS server [Suzuki 2005] 'dns.b.org'. The DNS server responds with the highest priority mail exchange server 'mx.b.org' for the domain 'b.org'. Sending server establishes SMTP connection with the receiving server and delivers the e-mail message to the mailbox of 'Bob' on the receiving server. 'Bob' downloads the message from his

mailbox on receiving server to local mailbox on his client computer using POP3 [Tzerefos 1997] or IMAP [Graham 1999] protocols. Optionally, 'Bob' can also read the message stored in his server mailbox without downloading it to the local mailbox by using a Webmail program.

Email tracking is a method for monitoring the email delivery to intended recipient. Most tracking technologies use some form of digitally time-stamped record to reveal the exact time and date that an email was received or opened, as well the IP address of the recipient.

Email tracking is useful when the sender wants to know if the intended recipient actually received the email, or if they clicked the links. However, due to the nature of the technology, email tracking cannot be considered an absolutely accurate indicator that a message was opened or read by the recipient.

Most email marketing software provides tracking features, sometimes in aggregate (e.g., click-through rate), and sometimes on an individual basis.

The Internet has smuggled itself to become a large part of most people's lives up to the extent that some people cannot go an hour's length without login in online to do you-know-what or to meet you-know-who. Some are even live online 24 hours a day. It may or may not be possible to track what an actual user does while online but it is these persistent people's online activities that makes it easier for someone, if he wants to, to track them and know actually what they have done, the sites they visited and even their individual preferences. These methods of tracking is usually done by e-commerce websites in order to build customer profiles and spam their mailboxes and emails with junk mails containing adverts of what they think that customer might be interested in.

Related Work Done

Muir Houston in 2008 explained that the purpose and focus of this paper is twofold. The first concerns the methodological issues involved with tracking a mobile population, namely students graduating from university; whose patterns of residence may for some time be transitory in nature. The second is to provide some details from a small sample of graduates on the issues raised in the transition from undergraduate study. The paper will tackle the two issues in order. First, an examination of recent methods of qualitative data collection which the development of information technology, in the form of Computer Mediated Communication (CMC), has allowed is undertaken. Second, the results of using one of these techniques (asynchronous e-mail) to gather information on the transition from undergraduate study are reported. In addition, advantages and disadvantages that can arise in the use of these new techniques are reported.

This paper engages with the overall theme of transition in a number of ways. First, the students are in a process of transition in status; from undergraduate to graduate. Second, in many cases, they may well be in transition in terms of location; from the university location either back home, or away to a new location. Third, they are in transition in terms of activity; either seeking or into employment or engaging in further study at a higher level. Finally, they may be in transition in relation to developing new social networks.

Michael Still et al. in 2011 explained that many businesses rely on email of some form for their day to day operation. This is especially true for product support organizations, which are largely unable to perform their role in the company if they're in boxes are flooded with malicious email, or if important email is delayed because of the processing of attack traffic. Simple Message Transfer Protocol (SMTP) is the Internet protocol for the transmission of these emails. Denials of Service (DoS) attacks are deliberate attempts by an attacker to disrupt the normal operation of a service with the goal of stopping legitimate requests for the service from being processed. This disruption normally takes the form of large delays in responding to requests, dropped requests, and other service interruptions. In this paper they explore the current state of research into Distributed Denial of Service (DDoS) attack detection, protection and mitigation for SMTP servers connected to the Internet. We find that whilst there has been significant research into DDoS protection and detection generally, much of it is not relevant to SMTP servers. During our survey we found only two papers directly addressing defending SMTP servers against such attacks.

Nasir Muhammad in 2014 explained that as much as the Internet helps to make life simpler, it does so much more to make it as insecure. This paper analyses some of the different methods how a user can be

tracked through the websites he visited while online and actually identify who that user is and associate him with a name, address and location, or even to go as far as knowing what he likes or dislikes. This paper starts by introducing the concepts of areas that will help to devise some various methods on identifying a web surfing person during a web session. It then went on to intertwine these concept with each other to actually formulate a method to make a user profile of a surfing individual and track what they have done. This paper continues with highlighting the implications of this tracking to a web user and the issue of privacy in the online world. The paper went on to finally conclude on some solutions of how a web user could actually protect him from being tracked due to his online behaviour.

M. Tariq Banday in 2011 explained that E-mail has emerged as the most important application on Internet for communication of messages, delivery of documents and carrying out of transactions and is used not only from computers but many other electronic gadgets like mobile phones. Over a period of year's e-mail protocols have been secured through several security extensions and producers, however, cybercriminals continue to misuse it for illegitimate purposes by sending spam, phishing e-mails, distributing child pornography, and hate emails besides propagating viruses, worms, hoaxes and Trojan horses. Further, Internet infrastructure misuse through denial of service, waste of storage space and computational resources are costing every Internet user directly or indirectly. It is thus essential to identify and eliminate users and machines misusing e-mail service. E-mail forensic analysis is used to study the source and content of e-mail message as evidence, identifying the actual sender, recipient and date and time it was sent, etc. to collect credible evidence to bring criminals to justice. This paper is an attempt to illustrate e-mail architecture from forensics perspective. It describes roles and responsibilities of different e-mail actors and components, itemizes meta-data contained in e-mail headers, and lists protocols and ports used in it. It further describes various tools and techniques currently employed to carry out forensic investigation of an e-mail message.

MATERIALS AND METHODS

There are three methods for tracking when an email is read. These techniques include:

1. Request a Confirmation Read Receipt
2. Append a Query String key to a Beacon Image
3. Implement a Http Module to Track Reads

Request a Confirmation Read Receipt

The first technique deals with requesting a confirmation read receipt. A confirmation read receipt is a special header tag that gets added to the email. When the email client (such as Outlook, Outlook Express, Eudora) reads the read receipt header, an email is generated and sent back to the sender. If you are using aspNetEmail, a code example for doing something like this is:

[C#]

```
EmailMessage msg = new EmailMessage( "mail.MyCompany.com" );
msg.FromAddress = "me@MyCompany.com";
msg.To = "you@YourCompany.com";
msg.ConfirmRead = true;
```

Although this technique is easy to implement, there are a number of problems with it.

First: is that not all email clients support the confirmation read receipt header. Thus, a user may open an email and read it, but you never get the read receipt email.

Second: of those email clients that do support it, most users have it turned off, because this is a technique used by spammers. Again, you will not get the confirmation emails.

Third: is that sometimes the user gets presented with a security dialog, stating the sender wants a read receipt. Some receivers view this as an invitation to their privacy, and will decline the dialog, and may unsubscribe from your newsletter.

Fourth: The Read Receipt would still need to be checked, and this would require an automated process scanning an inbox or a folder.

Append a Query String key to a Beacon Image

Another popular technique is to embed a transparent beacon image into the HTML text of the email. For example, your raw HTML email text may look like

<img alt="Company Logo" src="http://www.mysite.com/images/companylogo.gif">
Hi John Doe,
<p>Here is the status of your daily stock quotes.

... more data here

... more data here
<img width=1 height=1 src="http://www.mysite.com/images/transparent.gif?key=jdoe">

Although this technique is user friendly there are two main issues with it.

First: You have to parse the IIS logs. Although this can be done, sometimes it become a tedious and time consuming task.

Second: This is a known spamming technique. And because it is commonly used by spammers, more spam filters are preventing the request of beacon images. Therefore your email will get read, which is still good, but the the image request will not be recorded in the IIS logs.

Implementing a HttpModule to Track Reads

The last technique is more advanced, and more powerful to use than either of these techniques. This technique implements a HttpModule to intercept a request to an image embedded in your email, such as your company logo, and records that request to the database. But before we get to that part, let's talk about HttpModules for a minute.

What is a HttpModule?

A HttpModule is a component that can be plugged directly into the HTTP processing pipeline of ASP.NET. This makes HttpModules very powerful, because we can manipulate requests and responses as they enter or exit IIS.

Using the HttpModule

Code will intercept an image that will actually be named the email key. The email key is unique to each email, and somehow relates back to the receiver of the email. In the previous example, the key was appended as the query string. In this example, the image is actually named the key. Also, instead of a transparent beacon, this image will actually be our company logo. So, using the HTML text of the previous email, the HTML now looks like

<img alt="Company Logo" src="http://www.mysite.com/images/jdoe.aspx">
Hi John Doe,
<p>Here is the status of your daily stock quotes.

... more data here

... more data here

So when the email client requests http://www.mysite.com/images/jdoe.aspx, HttpModule will perform three functions. It will

1. Intercept that request
2. Extract the key 'jdoe'
3. Return the actual company logo, which is actually named "myRealLogo.gif".

Flow Chart for email tracking is shown in Fig.2. The steps for email tracking algorithm is as follows:-

Step 1. Compose the email.

Step 2. Send the hidden link of query string behind an image.

Step 3. The recipient opens the email.

Step 4. Information (date of email reading, subject, email ID) of the recipient send to the server page by query string.

Step 5. Check whether the information is currently exists for particular subject. If the information is currently exists for particular subject then the counter is updated by 1 in the database. Otherwise the information is stored in the database.

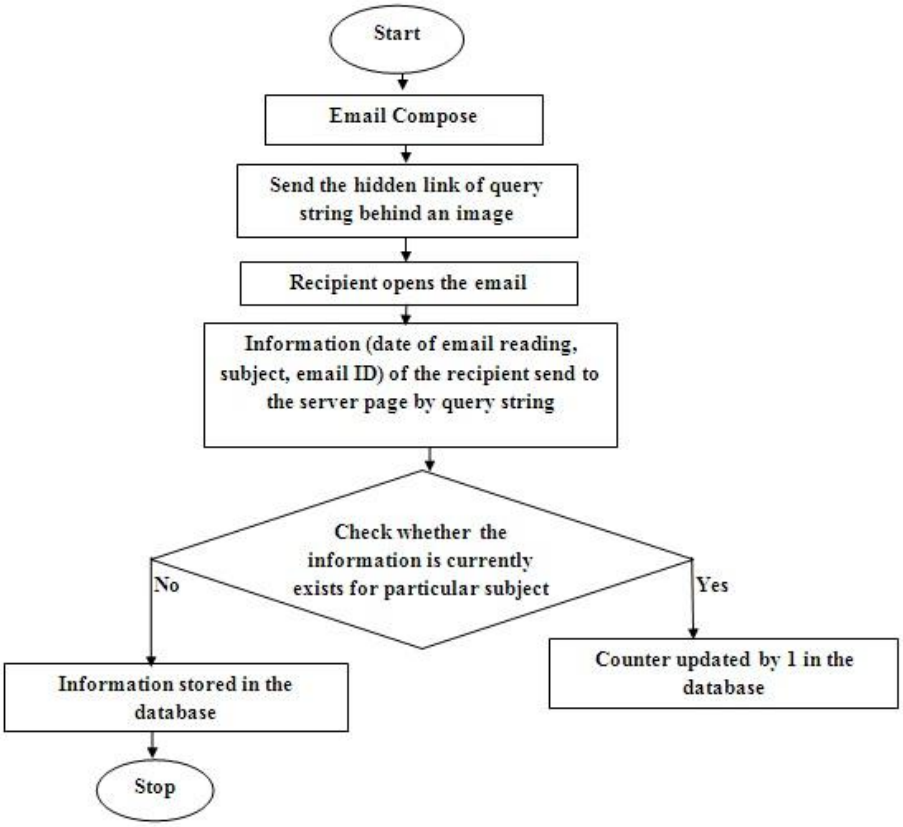

Figure 2: Flow Chart of the Purposed Method

RESULTS

LOGIN FORM FOR EMAIL TRACKING

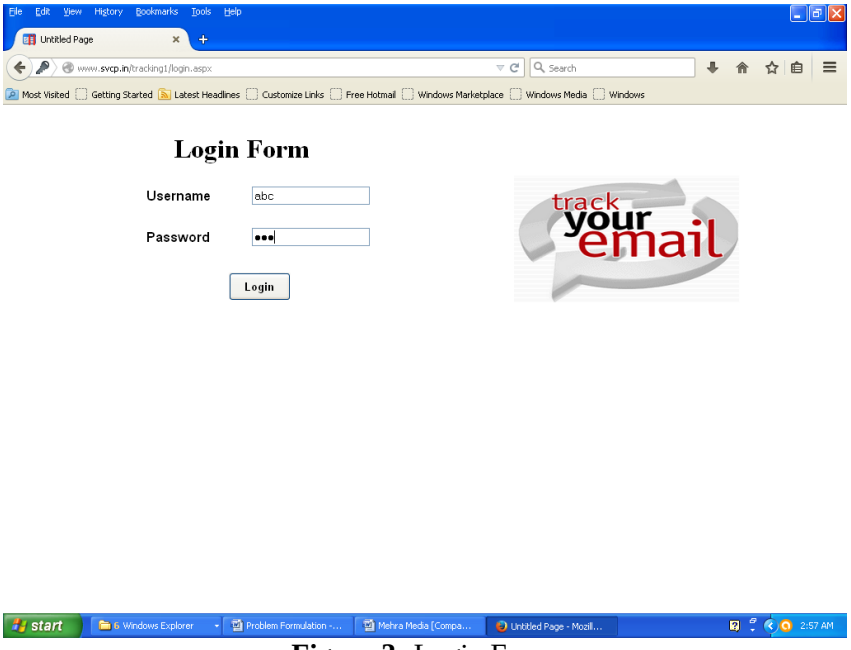

Figure 3: Login Form

Login Form authenticates the user, after entering valid username/password user can redirect on main form.

Main Form

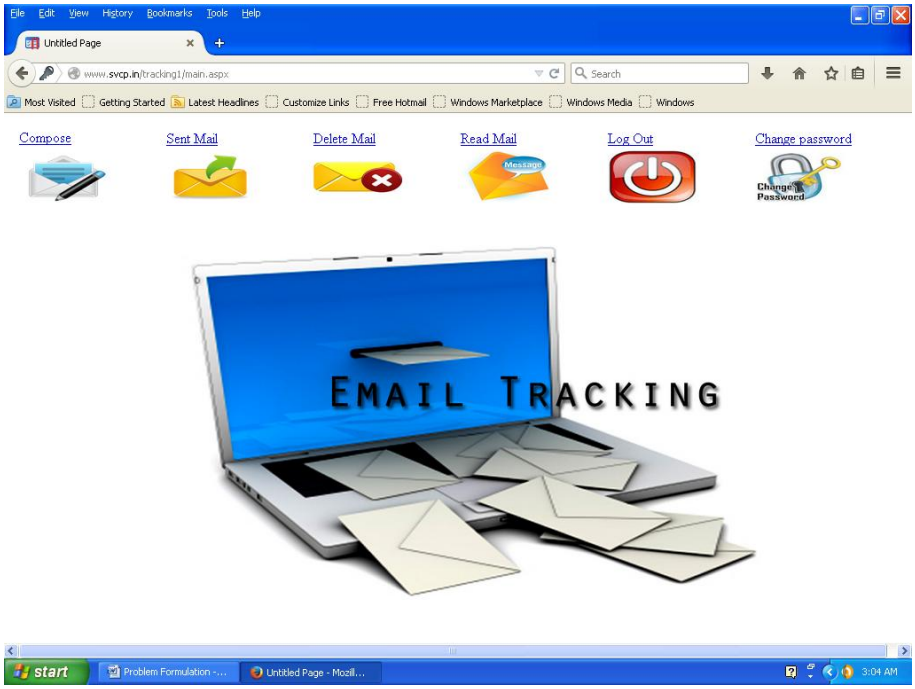

Figure 4: Main Form

As shown in Fig.4 Main form having all the options as Compose Email, View Sent Email, Delete Email, View Status of Read Mail, Log Out and Change Password.

Compose Email

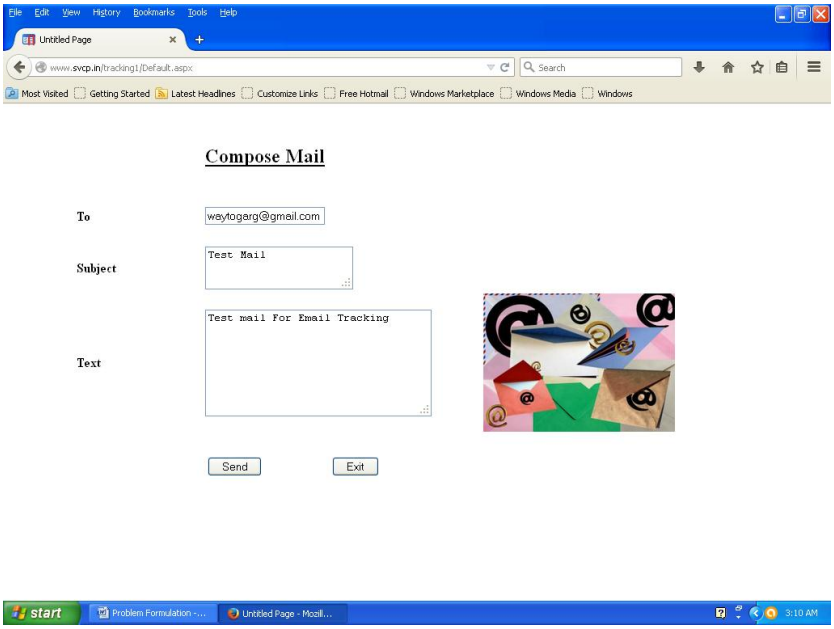

Figure 5: Compose email form

Compose Email form is used to draft a new mail to recipient having the options to enter the email id of recipient, subject and Text. When we click on send button mail goes to the recipient and display the message as **Mail Sent.**

View Sent Mails

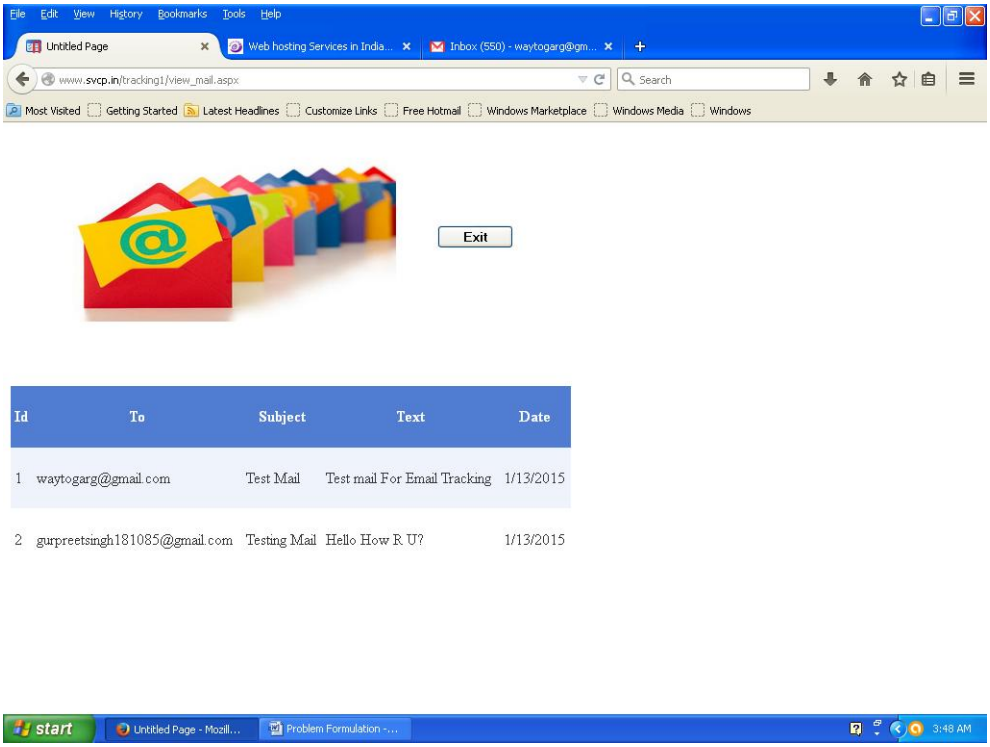

Figure 6: View sent mails

It displays all the sent mails by showing subject, Text and date.

Delete Mails

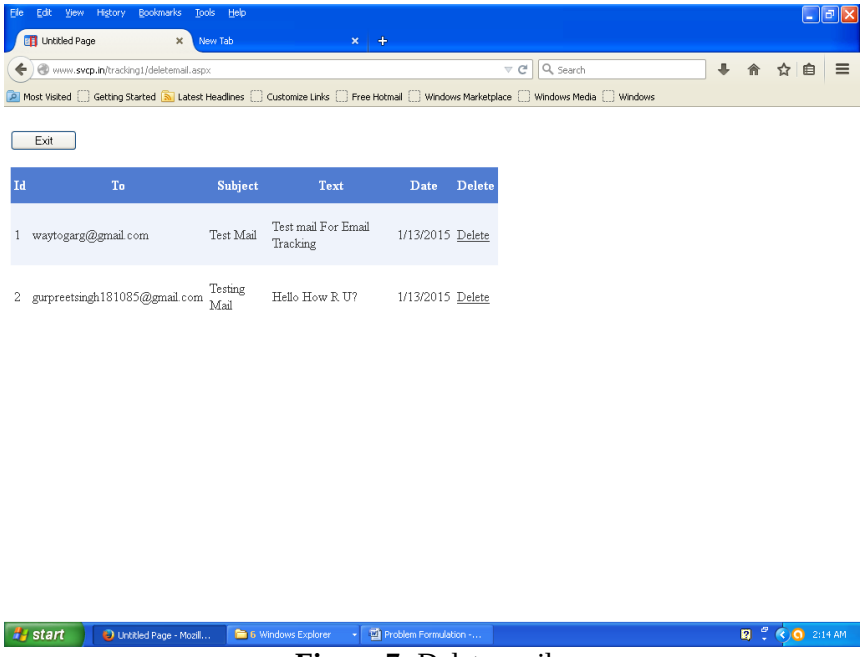

Figure 7: Delete mails

In this page user has option to delete the sent mails.

Read Mails

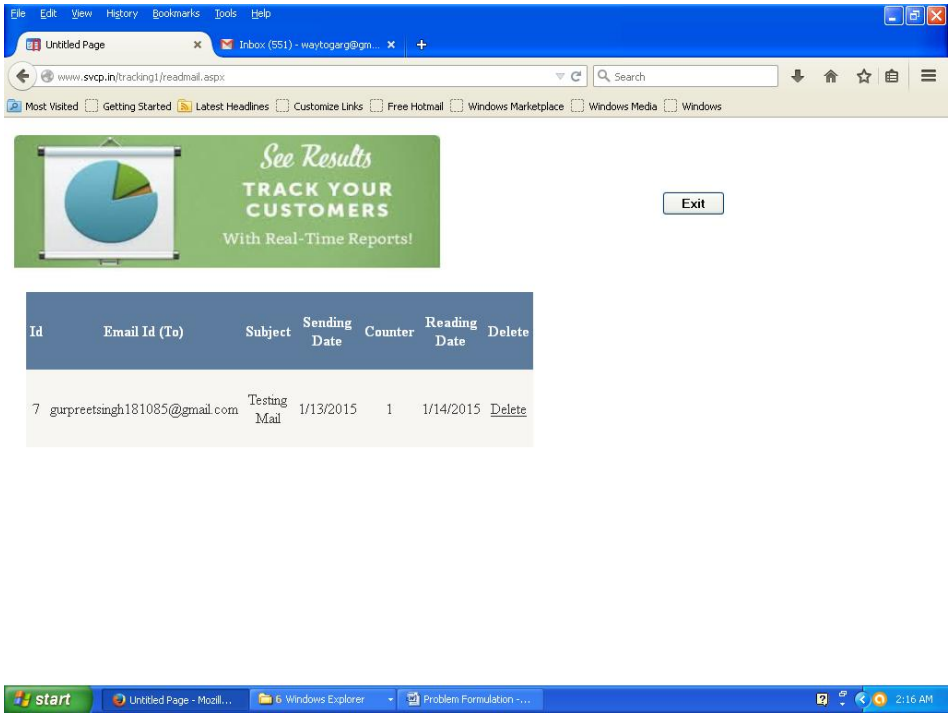

Figure 8: Read mails

This page shows the list of recipient who reads the email send by sender it shows Email Id, Subject, Sending date and reading date of recipient. In this print screen it shows gurpreetsingh181085@gmail.com read the email one time and waytogarg@gmail.com not read the email.

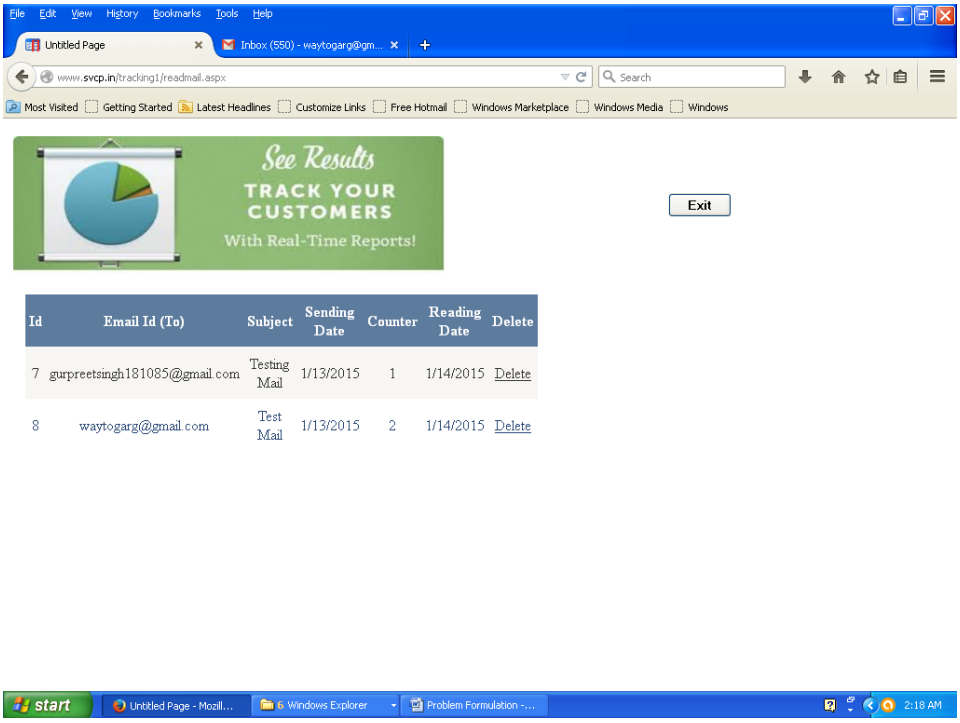

Figure 9: Counts Read email

Fig. 9 shows the list of recipient who reads the email send by sender it shows Email Id, Subject, Sending date and reading date of recipient. In this print screen it shows gurpreetsingh181085@gmail.com read the email only once and waytogarg@gmail.com read email twice.

Table: The email send and receiving results

S. No.	*Subject*	*Email Send*	*Receiving Results*
1.	*Test Email*	*100*	*89*
2.	*Wishes For happy Diwali*	*250*	*217*
3.	*Best Wishes For new Year*	*600*	*430*
4.	*Gift Coupons*	*3000*	*2930*

Fig. 10 shows this table results in graph form.

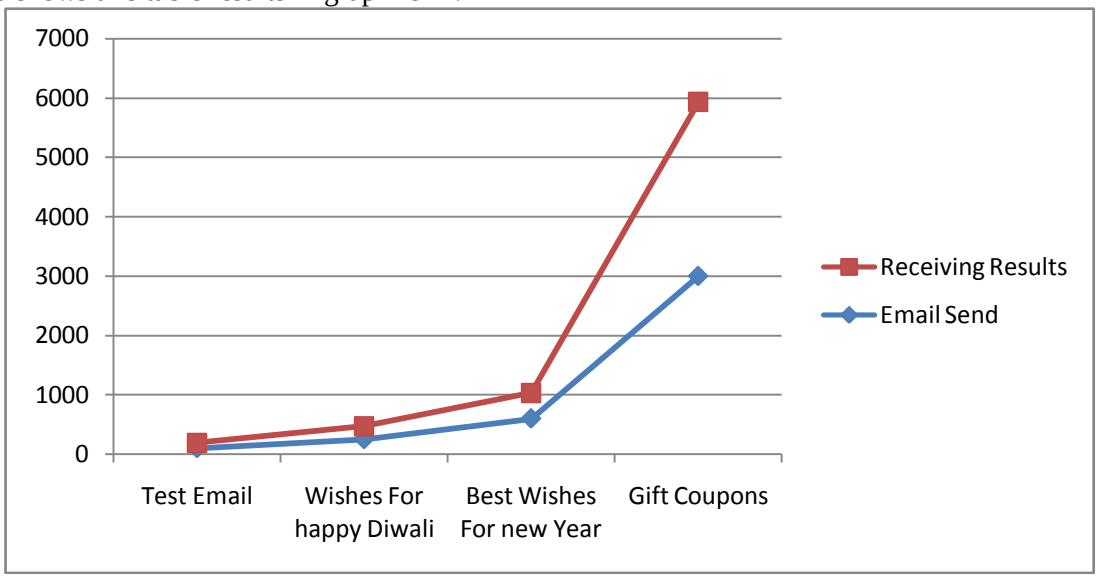

Figure 10: Graph shows the email send and receiving results

Conclusions & Future Scope

Some email applications, such as Outlook, employ a read-receipt tracking mechanism. The sender selects the receipt request option prior to sending the message, and then upon sending, each recipient has the option of notifying the sender that the message was received or read by the recipient.

However, requesting a receipt does not guarantee that you will get one, for several reasons. Not all email applications or services support read receipts, and users can generally disable the functionality if they so wish.

Email tracking is useful when the sender wants to know if the intended recipient actually received the email, or if they clicked the links. However, due to the nature of the technology, email tracking cannot be considered an absolutely accurate indicator that a message was opened or read by the recipient.

But this web portal help the user to identify that how many and which recipient read the send mail. With the help of this application marketing company/executives have exact figures to track open, click-through and conversion rates, making it simple to spot how a campaign can be improved.

Nowadays and in the future, application developers have to take into consideration the use of email tracing methods in order to efficiently implement their systems. In the future, application developers should consider using those methods for email tracing presented in this thesis, to implement their systems with full efficiency.

As the objective of the research is to provide an algorithm to solve the problem that the user has exact figures to track open, click-through and conversion rates, making it simple to spot how a campaign can be

improved. The query string method successfully works in implementation of email tracing. The sender gets confirmation when the recipient read email. In future developer may improve the efficiency of algorithm by adding the feature that sender gets confirmation when recipient delete or forward the email.

ACKNOWLEDGMENT
I would like to thanks my parents and my friends for their support and trust.

REFERENCES
Banday M Tariq (2011). Techniques and Tools for Forensic Investigation of E-Mail. *International Journal of Network Security & Its Applications (IJNSA)*, Vol 3(6).
Email tracking [online]. Available:http://en.wikipedia.org/wiki/Email_tracking
Graham J (1999). Enterprise wide electronic mail using IMAP. *SIGUCCS '99: Proceedings of the 27th annual ACM SIGUCCS conference on User services: Mile high expectations*.
Houston Muir (2008). Tracking Transition: Issues in Asynchronous E-Mail Interviewing. Vol 9 (2).
Still Michael & McCreath Eric C. DDoS Protections for SMTP Servers. *International Journal of Computer Science and Security (IJCSS)*, Vol 4(6).
Suzuki S, Nakamura M (2005). Domain Name System—Past, Present and Future. *IEICE Transactions of Communication*, E88b (3), 857-864.
Tzerefos Smythe, Stergiou Cvetkovic (1997). A comparative study of Simple Mail Transfer Protocol (SMTP), Post Office Protocol (POP) and X.400 Electronic Mail Protocols. *In Proceedings of the 22nd Annual IEEE Conference on Local Computer Networks*, 545-554.