On the Authentication of Date in E-mail using Trusted Time Stamping Service

M. Tariq Banday
P.G. Department of Electronics and Instrumentation Technology
University of Kashmir, India

ABSTRACT
In e-mail date spoofing an e-mail message contains forged date field that keeps e-mails listed on top in recipient's mailbox in some commercial and corporate e-mail servers, thereby maximizing the chances of immediate attention by its recipients. The date header field in a date spoofed e-mail may contain a date which is ahead or before the actual date it was sent and thus a date spoofed e-mail may be either a pre-dated or a post-dated message. E-mail date spoofing which emerged as a spamming trick can lead to manifold of problems like i) confusion to recipients, ii) loss of work productivity, iii) increase in false positive, iv) various time scheduling problems, v) increases chances of opening spam, vi) host legal issues, and vii) render date field insignificant. Date header field of an e-mail message is a trust field and can be violated without being detected by protocols involved in the e-mail system. To ensure credibility of dates in e-mail messages a trust mechanism can be devised by incorporating a trusted date and time signature at sending, transporting and receiving MTAs by the use of some designated third party Trusted Time Stamping Service (TTS). A TTS supports assertions of proof that a datum existed before a particular time. In order to associate a datum with a particular point in time, a Time Stamp Authority (TSA) may need to be used. This trusted third party provides a proof-of-existence of a particular datum at a given time. The TSA can also be used to indicate the time of submission when a deadline is critical, or to indicate the time of transaction for entries in a log. A digital time stamping service issues timestamps which associate a date and time with a digital document in a cryptographically strong way. The digital time stamp can be used at a later date to prove that an electronic document existed at the time stated in its time stamp. This paper proposes an effective measure employing the use of trusted third party time stamping service for authentication of date in e-mail messages. The model proposed can check and control date-spoofing at sending, forwarding, or receiving servers.

General Terms
E-mail, Spoofing, Time Stamping Service, Digital Signature, Authentication, Algorithms, Network Security, E-mail Security.

Keywords
E-mail Date Spoofing, Security Protocols, E-mail Security, Time Stamping, Trusted Time Stamping Service, TSS, Time Stamping Authority, TSA.

1. INTRODUCTION
In computers clock pulses synchronize events and system clock is used to keep track of current date and time. Date and time of activitiescarried out through or on these computers is recorded for various purposes. Date and time of a computer can be changed at a fly by user who has its control. In an e-mail message date and time is recorded in three fields namely date, resent-date and received header fields. The sender's computer and during transmission from sender's computer to recipient's computer while being handled by several intermediate nodes, servers use respective computerssystem clock to record date and time in it. A recent study has shown that spammers forge date field of an e-mail by either changing the system clock or by using deceptive e-mail client programs to maximize chances of immediate attention by recipients of spam e-mails. The forged date may either be before or ahead of the current date thus making a particular e-mail a post-dated or pre-dated message. A recent user survey by Banday et al in [1] reported that most of the users consider date of e-mail messages authentic and do not suspect forgery in it,though, a considerable number of e-mail users are aware of spam, sender spoofing and e-mail security protocols.

The remaining paper is organizes as follows: section 2 reports background study of e-mail date spoofing, in terms of tricks for sending it and subsequent handling by e-mail servers, its implications and proposed methods for its detection and control. Section 3discusses time stamping service and its working. Section 4 proposes a model for the control of date spoofing at sending, receiving and intermediate servers using time stamping protocol (TSP)which is followed by conclusion.

2. BACKGROUND STUDY

2.1. Submission and Reception of Date Spoofed E-mails
SMTP [2] is used for sending e-mail from the sender's client to the sender's server and for communication between sender's SMTP server and recipient's SMTP server. However, it is not used for accessing the recipient's mailbox. This is owing to a variety of reasons that include:

a) Incorporating multiple access protocol in SMTP would require additional functionalities leading to complexity of SMTP.

b) SMTP works on a push model as the transactions are initiated by the sender. For incorporating mailbox access functionalities, it needs to access to requests from recipient's clients. This would again make SMTP difficult rather than simple.

c) Incorporating mailbox access functionalities within SMTP would limit its flexibility for access using diverse technologies.

d) Further, this incorporation would also limit users to access e-mail from specific clients and thus shall pose severe difficulties to users who may need access to their mailboxes from different clients in different parts of the Globe.

RFC 1733, "Distributed Electronic Mail Models in IMAP4", describes three different paradigms, or models for mail access and retrieval. These are online access, offline access and disconnected access models.In online access users have direct online access to their mailboxes. However, this is possible only if users' machines are connected permanently to the Internet and are configured as SMTP servers. This is impractical baring some special examples like for users who run their own SMTP servers.In offline access user's client computer establishes a connection to the server where his mailbox is located using some protocol. The mail is downloaded from the user mailbox on the server to the mailbox on the user client and the mail is deleted from the user's server mailbox. Processing of the mail is performed on the user's client computer and thus does not require a continuous connection to the Internet. Disconnected access is a hybrid of online and offline access models. Users download mails from the server, and manipulate them without requiring a continuous connection to the server. However, the mail is not deleted from the server, as in the offline model. Periodically, users connect to the server and synchronize mailboxes on their server and client computers. Online access has the main benefits of instant speed and universal access from any location. But requires a user to be online and often requires UNIX e-mail clients. Offline access has the main advantages of simplicity and short connection time requirements. However, this method is inflexible and poorly suited to access e-mail from different machines. Still, it is currently the most popular access method because simplicity is important; it is best typified by POP. The advantages of this method are: ability to access mail quickly and use offline mail processing. In recent years, a somewhat new mailbox access method has become popular: e-mail access using the World Wide Web. This technique allows a user to access his mailbox from any computer with an Internet connection and a web browser.

2.1.1. Submission

E-mail message is submitted from the client computer of the sender to sending SMTP server using a client e-mail program e.g. MS Outlook Express, MS Office Outlook, Eudora, etc. which forwards this message to the SMTP server of the recipient. A web server in case of web based e-mail service e.g. www.inbox.com, www.mail.com, etc. can also be used to send e-mail to its recipients. Web based e-mail services have advantage of easy setup and use from any computer but are inflexible as users do not have direct control on their mailboxes and users must be online for composing and reading mail.

Date spoofed e-mail can be submitted from the sender's client computer to the sender's SMTP server by the use of any e-mail client program and altering the system clock of the client computer or by manipulating the send date field within the client program. Following programs and tricksmay be used [3]:

a) Running custom e-mail programs or bulk e-mail tools called e-mailers and directly manipulating the send date header field while sending the e-mail message.

b) Running well know e-mail programs e.g. MS Outlook, Eudora, etc. on client computer with manipulated system clock before sending the e-mail message.

c) By establishing a direct connection with the receiving SMTP server through network utilities like Telnet and sending e-mail messages with spoofed date header fields.

Since sending servers do not authenticate date header field of an e-mail message or check its correctness unlike some other header fields which are authenticated by protocols like DKIM, SPF, etc., date spoofed e-mails are transmitted to receiving SMTP servers. Commercialweb based commercial e-mail services which provide online access to mailboxes use date from the system clock of the Web servers and not from the client computers and thus cannot be tricked to send date spoofed e-mail messages. However, any web based e-mail service can be created that can accept user defined date for e-mail messages.

2.1.2. Reception

After being submitted from e-mail client by the sender to his SMTP server, the sender's SMTP server transmits it to its destination which may pass through many intermediaries like routers, and mail servers before reaching the recipient's SMTP server.With a few exceptions, currently, all receiving SMTP servers do not check the correctness of date in incoming e-mails and thus allow date spoofed e-mails to be stored in the recipient's mailboxes. An experimental study in various aspects of date spoofing by Banday et al in [1] of various commercial and corporate e-mail servers has reportedthe following findings about date spoofing and its handling by e-mail servers:

a) Most of the e-mail servers whether corporate or commercial accept date spoofed e-mails. However, some reject post-dated e-mails ahead of the current date by two days.

b) Post-dated e-mails remain listed on top in the inbox if sent date field is used as a sorting field instead of received date which is the case with some commercial web based e-mail services.

c) Short date formats used in some e-mail programs along with send date as a sorting field can make it difficult to suspect a mail being spoofed in date without carrying out extensive header analysis.

d) Currently spam filters and e-mail security protocols do not check for correctness of send date in e-mail messages.

e) Only a small percentage of e-mail users are aware of date spoofing and little is known about theproblems it can spawn.

It has been reported that most e-mail servers accept incoming date-spoofed e-mails, however, different web mail programs and e-mail client programs list e-mail differently. Some sort the list of new e-mails in the inbox on send date while others on received date. Mail retrieval protocols POP3 [4] and IMAP4 [5] do not check the integrity of send date field and thus allow E-mail client programs to retrieve date spoofed e-mails form recipient's SMTP server to their local inboxes.

2.2. Implications

Date spoofed e-mails may be just spam e-mails merely to advertise different types of goods, services or ideas, or trick

recipients into giving up their credentials, or cause a temporary crash of a mail server or to gain immediate or special attention of recipients. Such a spam mail will only waste time of users opening it, waste Internet resources, and consumes storage and bandwidth[6].However, date-spoofed e-mails whether pre-dated or post-dated can cause various other problems to the recipients or recipient's organization that include [1]:

a) Spam e-mails spoofed in date can further add to the confusions and also increase the chances of false positive of spam filters.

b) It can also create a complex problem in scheduled activities. It is universally known fact that e-mail is being used not only in private communications but also in business communications wherein tenders, bids, evaluation reports, RFP submissions and numerous similar scheduled activities where a response within the stipulated time is required are carried out. E-mail programs that sort their e-mails by receiving date but accept e-mails in spoofed-date although save their recipients from confusion but at the same time it can result in more complex problems in situations where an e-mail pertaining to something is inacceptable before or after a particular date.

c) An e-mail message is first send from sender's client to sender's SMTP server which forwards it to the receiver's SMTP server and in between it travels through various transporting MTA's which may not immediately deliver or transport an e-mail message due to some fault or their policy. It is also possible that a sending MTA or transporting MTA whose clock is not correctly set can insert a wrong date in the received field. Thus besides send date field received date field may also be incorrect in an e-mail message. In absence of a firm technical standard for determining the correctness of both send and received date fields, the parties can contest the correctness of date in case of a dispute especially in time scheduling activities which can result in an extended legal battle.

d) All date related fields in an e-mail message are trust fields and this trust can be violated due to an error or deliberate trick by any entity involved in the e-mail system rendering all date related fields namely date, resent-date and received header fields insignificant.

2.3. Detection and Control

Security protocols and procedures [7] to secure insecure SMTP developed over a period of years do not permit detection or control of date spoofing. Highly secure anti-spoofing protocol S/MIME] that uses digital signatures [8] to digitally sign an e-mail message, the signature date and time is obtained from the system clock of the signer's computer which can easily be spoofed. Most widely used security protocol DKIM which also uses digital signatures signs various sender information fields which may also include date field but no standard method has been suggested for such sigining of date field in this protocol and mostly date field is not signed in the signature process. Security procedures which include diverse anti-spam procedures do not check for the correctness of date in an e-mail message.

2.3.1. Detection

Detection of date spoofed e-mail messages involve determining forged or incorrect date in date, resent-date and received header fields. This though will not stop transmission of such e-mail messages but will enable forensic examination of suspected e-mails messages.As detailed in [1] extensive header analysis of suspected e-mail messages can be carried out to determine possible date forgery.

Date header set by originator of e-mail, Resent-Date header set by any mediator and Received header set by originator, relay, mediator or destination are the three header fields that contain dates in an e-mail message. An e-mail message has one date header, none or more resent date headers, and none or more received header fields. Date field, unlike as to how it is understood nowadays, is not an indicator for date and time of actual transport but instead as per RFC2822 [9]it should contain the creation date of the e-mail message. However, received filed also called trace field besides others records timestamp of its arrival at respective mail transfer agent. The date and resent-date in their current context mean the date of transport of message or in their true context which means the date of creation of message cannot be authenticated unless some control is put on them. In their current context, an algorithm given in [3] may be used to calculate the date difference between dates found in the e-mail message and the current system date of the receiving agent giving priority to received field followed by resent field and date field. If the difference is greater by some margin, the e-mail message may be reported as spoofed in date.

2.3.2. Control

The date and resent-date fields in their current context which means the date of transport of message or in their true context which means the date of creation of message cannot be authenticated unless some control is put on them. Further, dates in the trace information cannot be taken as authentic unless system clock of source, intermediate and destination mail agents are certified to be correct. Banday et al in [1] and Banday in [3] have given the following directions or techniques for the control of date spoofing.

a) Not to use the send date and resent date fields and not to trust them at all in e-mail system. In this case the date and time from the clock of the first or the last MTA may be used and trusted. This will however, result in losing the significance of date fields in e-mail message which is thus not a practical solution.

a) Every MTA will check the correctness of date and time of the arriving e-mail message by comparing the send/resent date with their system clock. In the difference is large e-mail may be discarded. However, this option requires time synchronized among all servers. This technique has been discussed in [1].

b) Both send and received dates are used in e-mail client programs. This however, will not stop transmission of date spoofed e-mail messages and will also not certify their correctness.

c) Make sigining of date header in DKIM protocol mandatory. In such casesince the date and time while sigining is taken from clock of the sigining computer which is not a trusted source.

d) Make use of some trusted time stamping service to put a trusted date/resent date in the e-mail message.A model for such a solution is discussed in the next part of this paper

3. TIME STAMPING

It is essential to record accurate and reliable time on electronic documents like a contract, a log file, a tender, an e-mail message, etc. before distributing, storing or transporting to ensure correctness of time and to avoid any possible legal issue. Organizations can use time to solve control and forensic issues by meeting the following two requirements: i) accurately timestamping of events and ii) synchronization of events to an accurate clock [10]. First requirement is satisfied if the event in the past actually happened at the instant it is alleged to have happened. A Time Stamping Protocol (TSP) ensures accurate and reliable timestamping using a Time Stamping Authority (TSA). Time stamping becomes a key feature when legal use of electronic documents with a long lifetime is required. The second requirement is satisfied if the events are synchronized to an accurate time index i.e. events happen at the instant they are intended to happen. TSP and synchronization both depend on a highly reliable and accurate time source but have different requirements for implementation. For TSA, access is over Public Key Infrastructure (PKI).

Time Stamping Protocol (TSP) is described in RFC 3161 [11] which has been updated by RFC 5816 and RFC 5544. TSP ensures accurate and reliable time stamping using a TSA which creates and sends a time stamps for a requested digital document whose hash is send by the client to the TSA. The time stamp is merged by the client with the document. Two types of translations happen in TSP and are the time stamp request generated by the requesting entity and the time stamp response generated by the requested TSA. TSP defines the format of transactions taking place between the client and the TSA and provides some suggestions for their transport.

Time stamping establishes evidence indicating that an electronic document was created or existed before a particular time [11]. An authority issuing such timestamps called Time Stamping Authority (TSA) may be established within an organization for internal use or may be operated as some Trusted Third Party (TTP) service. A trusted timestamp becomes an irrefutable proof that a document existed at a particular time and it thus ensures accuracy, integrity and binding of time and the digital document. As per RFC 3161 a TSA has many requirements that are:

a) Use of a trustworthy source of time,

b) Inclusion of a unique integer for each time stamp token generated,

c) Only a hash representation of the datum should betimestamp,

d) Production of timestamp token on the reception of a valid request whenever possible,

e) Inclusion of an identifier to uniquely indicate the security policy under which the token was generated,

f) Examination of OID,

g) Non examination of imprint other than length of the imprint being time stamped,

h) Non-inclusion of any identification of the requestor,

i) Sigining of time stamp token using an exclusive key meant for this purpose, and,

j) Inclusion of some other information in the token only if requested by the requestor. There are many service providers that provide third party trusted time stamping services for competitive rates which can timestamp diverse range of documents.

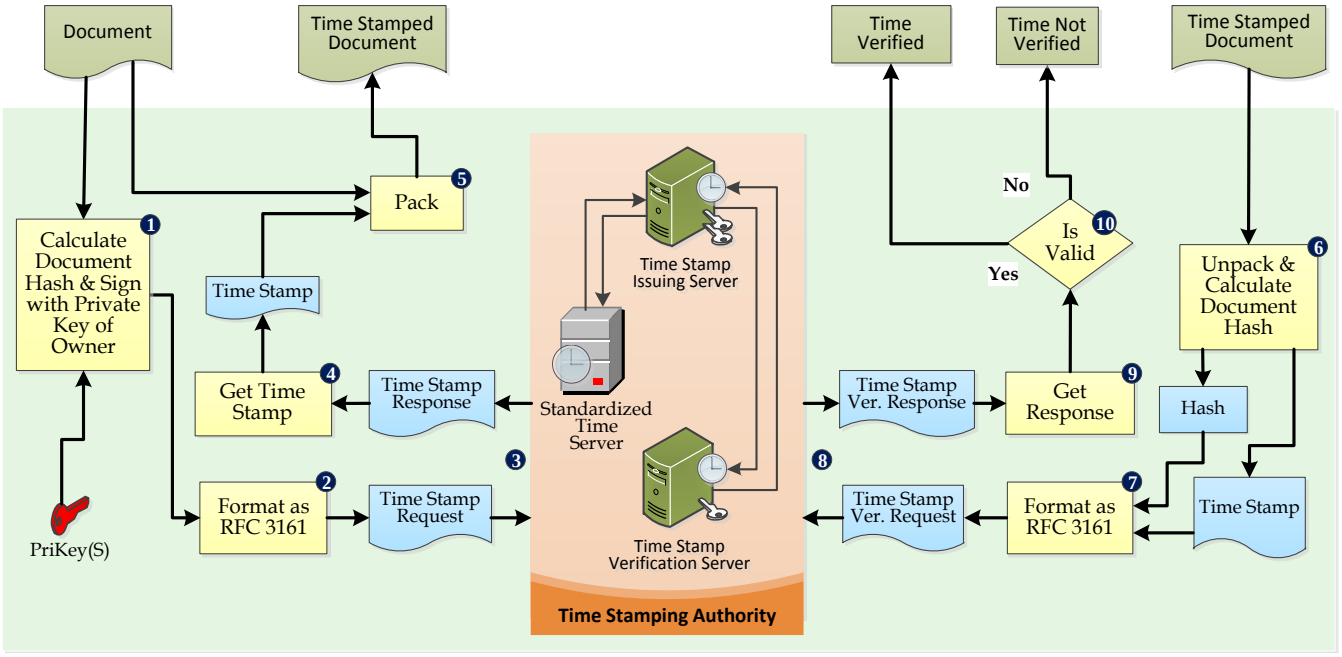

Figure 1: Time Stamping Process

3.1. Working of TSP

The working of TSP is illustrated in figure 1. A hash of the document to be time stamped is generated and signed with the private key of the document owner in step 1 which is formatted as RFC 3161 time stamp request in step 2. It is submitted to a trusted time stamping authority in step 3, which generates a time stamp response containing the time stamp signed by its private key in step 4. The time stamp is issued to the requestor in a time stamp is send to the requestor and is also stored in the database of TSA. The time stamp is packed with the document, in step 5 to form a time stamped Document. An online or offline process is used to check the correctness of data/time of a time stamped document. In online verification process, verification is performed by the TSA and in offline method; it is done on the client itself.

In online verification, time stamped document is unpacked to get the time stamp in step 6 which is formatted as RFC 3161 format in step 7 to produce a time stamp verification request which is submitted to the trusted time stamping authority for verification in step 8. The time stamping server replies with a time stamping verification response containing the time validation in step 9. The date and time of document is verified in step 10 as correct in case the time validation response is positive and not otherwise. During time stamping process some other operations including verification of the digital signature of TSA in the time stamp token before using it is done to determine authenticity of the TSA. Similar process is also used during verification operation. Various offline and online variations for the verification of time stamps have been proposed in [12].

4. TIME STAMPING FOR CONTROL OF DATE SPOOFING

Recent events have shown that sometimes cybercriminals tamper date/time of e-mail messages to commit crime or to cover them up. A trusted time stamp on an e-mail message can in these situations be used to ascertain the authenticity of date/time. Further, to use date and resent date header fields in their true contexts (time of transport) it is essential to time stamp e-mail messages.

An e-mail message is created on a client computer using an e-mail client program which submits it to the sender's SMTP server for onward transmission. Next, the e-mail message is transmitted by the sender's server to the recipient's SMTP server which before reaching its destination is handled by several intermediate nods. The current system date/time of the client computer is placed in the date or resent date field by the client program before submission to sender's SMTP server. In case of webmail services which provide access to the inbox of the sender's SMTP server through a webpage the date/time is from the clock of the server itself. Every intermediate node handling the e-mail message may add a trace information containing the date and time in the received header field.

Steps involved in the use oftime stamping for control of date spoofing in e-mail are illustrated in figure 2 and the operations performed are described below

a) Composing:

An e-mail message is composed by the sender on a client computer using an e-mail client program like MS Outlook, etc. or by using a web mail program. Required header fields like the recipient's e-mail address, subject, etc. are filled in by the sender and some header fields are filled in by the e-mail client program including date which may be spoofed by the sender. But its body is completely composed by the sender in a particular format. Step 1 in figure 2 shows this operation.

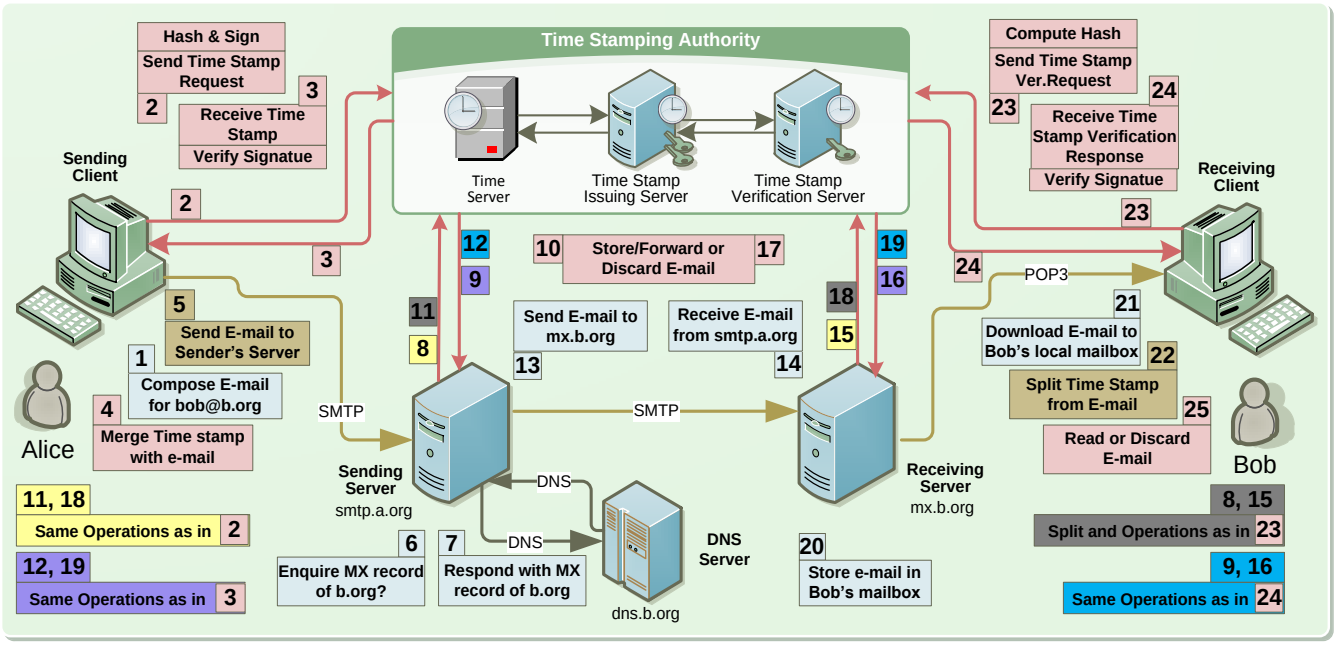

Figure 2:E-mail transmission and reception employing time stamping

b) Time Stamping:

Steps 2, 3 and 4 are used to time stamp the e-mail message. In step 2 a hash of the e-mail message is computed on the client computer and signed using the private key of the sender which is formatted as RFC 3161 time stamp request. This request is send to a TSA which generates a time stamp response containing the time stamp. The received time stamp is merged with the e-mail as a custom header field TimeStamp/text to meet the Multi-Purpose Internet Mail Extensions (MIME) [13] requirements of the e-mail message. MIME uses special techniques to encode different types of media into ASCII text form, such as graphical images, sound files, video clips, applications programs, compressed data files, and many others. MIME discrete media types allow MIME to represent hundreds of different kinds of data in e-mail messages.

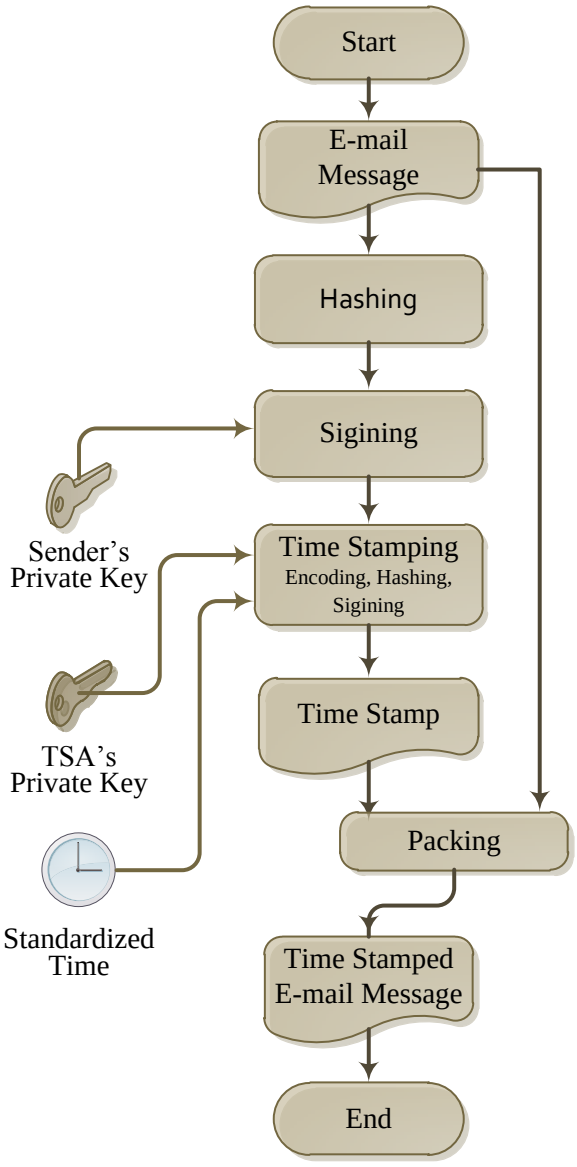

Figure 3: Steps involved for time stampingof an e-mail message

A unique custom MIME field may be created to support distribution of time stamp with e-mail message. In case an e-mail message is composed using a web mail interface, the webmail program performs the operations of time stamping at the sender's SMTP server in steps 11 and 12. In addition to time stamping in steps 2, 3 and 4, SMTP server of sender and receiver also time stamp the e-mail message in steps 18, and 19. Further, any intermediate SMTP server can put its own time stamp on the e-mail message in case e-mail delivery is indirect. The e-mail may accept any number of custom TimeStamp/text header fields that can be merged with the message. Every new time stamp on an e-mail message will authenticate the last time stamp as every handling node shall time stamp the message only if it finds no spoofing of the date/time in it. This timestamp/text header shall be similar to received header but unlike this header the time and date is in the form of time stamp generated by a trusted third party time stamping server. The flowchart in figure 3 shows the process of time stamping an e-mail message at the sender's client, intermediate servers, and at recipient's client including the steps performed at TSA.

A hash of the message to be time stamped is created using some hashing algorithm which is then signed using the private key of the sender. The public key of the sender is merged with the hashed and signed message signature (not shown) which is submitted to the TSA for generation of time stamp. The TSA encodes the time stamping data i.e. message signature, date/time, etc. called time stamp information structure using DER encoding to convert it into a sequence of bytescalled encapsulated content information for the purpose of packing and transmitting. A highly reliable clock synchronized by international time authority and national measurement institutions is used as a source of time. The encapsulated content information is hashed to produce signed time stamp information structure which is DER encoded, hashed and then signed by the private key of TSA. The resulting signature is appended to signed attributes to create signer information.Different steps in the process of forming a time stamp are given in [14]. The time stamp is next packed with the message to produce a time stamped e-mail message.

c) Mail Submission:

The e-mail message after being time stamped in step 4 is submitted to the SMTP server of the sender for transmission to its destination. Sending server performs a lookup for MX record of receiving server specified in the recipient e-mail address in the DNS Server in step 5 which it responds with the MX record of receiving MTA in step 6. After performing the steps 8, 9 and 10 discussed in the following section and steps 11 and 12 discussed in above section, the sending server may initiate SMTP transaction with the receiving serveror some intermediate SMTP server in case of non-direct delivery and sends time stamped e-mail to the receiving server.

d) Time Stamp Verification:

Verification of time stamp can be carried out at nodes handling the e-mail message i.e. at sending SMTP server, forwarding SMTP server, receiving SMTP server, and the receiving client. Verification of time stamp may be performed by TSA or at the client itself. Online verification of time stamp is carried out by unpacking the time stamp from the received e-mail message, computing hash of the message and submitting a time stamp

verification request to the TSA in step 8 and 15 at the sending and receiving servers and in steps 22 and 23 at the recipient's client. Time stamp verification replies are received from the TSA in step 9 and 16 at the sending and receiving servers and in step 23 at the recipient's client. E-mail message. On the basis of the verification replies received from the TSA, the sending, forwarding and recipient servers may either store/forward or discard the e-mail message in steps 10 and 17. A user or his client may decide to read or discard the e-mail message on the basis of the time stamp versification response in step 24. Verification of the time stamp at sending, forwarding, or receiving servers before forwarding or storing the e-mail with its time stamp will provide an advantage of possible checks for date spoofing at intermediate handling nodes and an accurate and authentic date/time information during the entire path.

5. CONCLUSION

Recent events have shown that sometimes cybercriminals tamper date and time of e-mail messages to commit crime or to cover them up. Thus, the date, resent date and date in received fieldsof an e-mail message in their current form cannot be trusted. Currently, e-mail server and client programs do not offer any control for date spoofing and techniques suggested for its control are either inadequate or suffer from some potential disadvantage.A model using time stamping protocol for the control of date spoofing at every handling node has been proposed in this model. The use of timestamping an e-mail message can ascertain the authenticity of date and time it was created, submitted, transmitted by a particular server and received by the recipient. Verification of the time stamp at sending, forwarding, or receiving servers before forwarding or storing the e-mail with its timestamp will provide an advantage of possible checks and control for date spoofing at intermediate handling nodes. Further, an accurate and trusted date and time information will be available within the e-mail at its destination.

6. REFERENCES

[1] Banday, M.T., Mir, F.A., Qadri, J.A., Shah, N.A. 2011. Analyzing Internet E-mail Date Spoofing, Journal of Digital Investigation, vol. 7, pp. 145-153, DOI:10.1016/j.diin.2010.11.001, URL: http://www.sciencedir ect.com/science/article/pii/ S1742287 610000812.

[2] Klensin. 2001.Simple Mail Transfer Protocol. IETF RFC 2821, URL: http://www.ietf.org/rfc/rfc2821.txt.

[3] Banday, M.T. 2011. Algorithm for Detection and Prevention of E-mail Date Spoofing. International Journal of Computer Applications, vol. 21(.6), pp. 7-11, DOI: 10.5120/2518-3421, URL: http://www.ijcaonline.org/ volume21/number6/pxc3873421.pdf.

[4] Tzerefos, P.,Smythe, C.,Stergiou, I.,Cvetkovic, S., 1997. A comparative study of Simple Mail Transfer Protocol (SMTP), Post Office Protocol (POP) and X.400 Electronic Mail Protocols.Local Computer Networks, Proceedings on 22nd Annual Conference on 2-5 Nov. 1997 pp. 545 – 554.

[5] Crispin, M. R. 2003. Internet Message Access Protocol - Version 4 rev 1. IETF RFC 3501.

[6] Mir, F.A., Banday, M.T. 2010. Control of Spam: A Comparative Approach with special reference to India. Journal of Information Technology Law, vol. 19(1), pp.22-59, DOI: 10.1080/13600831003589350, URL: http://dx. doi.org/10.1080/13600831003589350.

[7] Banday, M.T., Qadri, J.A. 2010. A Study of E-mail Security Protocols.eBritian, British Institute of Technology and E-commerce, Issue 5, Summer 2010, pp. 55-60, URL:http://www.bite.ac.uk/ebritain/ebritain_summer_10.p df.

[8] Banday, M.T. 2011. Easing PAIN with Digital Signatures. International Journal of Computer Applications, 29(2), pp. 46-56, DOI: 10.5120/3533-4822, URL: http://research .ijcaonline.org/volume29/ number2/ pxc3874822.pdf.

[9] Resnick, P., Ed. 2011. Internet Message Format. IETF RFC 2822, URL: http://www.ietf.org/rfc/rfc2822.txt.

[10] Cronk, R. 2003. Time: The Currency of Computer Crime, Symmetricom, URL: www.greatwriting.com/ ABOUT_ DOWNLOADS/forensics.pdf.

[11] Adams,C., Pinkas, D., Cain, P., Zuccherato, R. 2001. Internet X.509 Public Key Infrastructure Time Stamp Protocols (TSP). IETF RFC 3161, URL: www.ietf.org/rfc/rfc3161.txt.

[12] Tsutomu, M., Tadahiro, S., Keisuke, I.2007. Time stamping system for electronic documents and program medium for the same. US Patent No: 7266698, URL: http://www.patentgenius.com/ patent/7266698.html.

[13] Ramsdell, B., Ed., 2004, Secure/multipurpose internet mail extensions (S/MIME) version 3.1 message specification. RFC 3851, URL: www.ietf.org/rfc/rfc3851.txt.

[14] Axelle, A., Vincent, G. 2002. XML Security Time Stamping Protocol, Cover Pages. URL: http://xml.coverpages.org/ApvrilleTStamp.pdf.