Impact Factor (SJIF): 3.632

International Journal of Advance Research in Engineering, Science & Technology

e-ISSN: 2393-9877, p-ISSN: 2394-2444

(Special Issue for ITECE 2016)

Email Based Trust Management System

Alpesh Patanwadia[1], Bhaumik Macchi[2]

[1]CSE, SLTIET College,Rajkot
[2]CSE, SLTIET College Rajkot.

***Abstract* — Trust management has received a lot of attention recently as it is an important component of decision making for electronic commerce, Internet interactions. Email based trust management system prevent the email based fraud. In this paper I am going to explain the trust management approach for the e-mail based environment encryption system using digital signature. Simple Mail Transport Protocol is the most widely adopted protocol for e-mail delivery. However, it lacks security features for privacy, authentication of sending party, integrity of e-mail message, nonrepudiation and consistency of e-mail envelope. This digital signature system takes into account the device limitations and thus generates a functional digital signature. Email is encrypted with the public key of receiver and decrypted by receiver's private key & the email is sign by digital signature so that email server can accept that email.**

Keywords-component; Email Based Trust management System, Email Security, Digital Signature Based Trust management system

I. INTRODUCTION

As the use of email is increases, risk associated with email is also increases. Email security is the big challenge among the researchers. Simple Mail Transport Protocol (SMTP) [1] was originally designed for a smaller community of users which was assumed to be well behaved and trust worthy. As such no heed was paid towards incorporating security protocols in it. But with its growth, this trust was breached, owing to lack of adequate security mechanism in it

In this paper I have introduce a way by which the email based ford can be caught by filtering the email based on digital signature which is sign with the public key of receiver and then send to receiver at the receiver and decrypted using private key of the receiver this key are maintain by the trusted authority by the role based trust management system here we can trust the authority because it is trust worthy by the certified authority and their digital signature can also be certified.

Add-on security protocols are widely adopted measures to provide security in e-mail systems. A Review of prominent add-on security protocols along with their working has been carried out in [3]. These protocols either use cryptographic techniques or encryption or some domain validation standards. A detailed survey of e-mail servers in dealing with problem of date spoofing and apprizing e-mail user behavior with regard to date spoofing has been carried out However, this study has not carried out study pertaining to sender spoofing and treatment of such e-mail messages by e-mail servers. The remaining paper is organized as follows: Section 2 introduces e-mail security and enlists security issues of SMTP. Section 3 describes limitations of the e-mail security protocols Intrusion detection functions include [2]: Section 4 analyses e-mail servers of some Commercial E-mail Service Providers. It also presents possible approaches to improve their efficiency. Section 5 appraises e-mail user practices, their knowledge of security protocols and also evaluates their confidence in e-mail system through a study which is followed by conclusion.

II. SECURITY ISSUES WITH SMTP SERVERS

E-mail messaging, security can be defined as the ability of the system to provide.

The theme is a common one in TCP/IP: a lack of security in how a protocol is implemented. And this all goes back to a common root cause: most of these protocols were developed when the "Internet" was just a small group of machines controlled by individuals who mostly knew and trusted each other, or who were able to use physical security means [25]. Developers never imagined TCP/IP being used by millions of anonymous "average Joe" users around the world, which necessitates far more attention to security than a small research internetwork like the ARPA net.

When it comes to SMTP, security matters are if anything worse than they are with the other protocols I mentioned above. Not only does SMTP not have any real security mechanism, the original relying model of SMTP communication is entirely designed around the idea of "cooperation" and "trust" between servers [25]. Since most SMTP servers would be asked to handle a certain number of intermediate transfers, each server was required to accept mail from any originator to be delivered to any destination.

The basic assumption in this model is that SMTP servers would all be "well-behaved", and not abuse the system by flooding intermediate servers with lots of mail to be delivered, or sending bogus messages to cause problems. This all changed as the Internet exploded in popularity in the 1990s. Con artists, hackers, and disreputable salespeople all discovered that e-mail could be used for "free" delivery of messages simply by submitting them to an SMTP server for delivery. The result was overloaded servers, primarily due to the sending of large quantities of unwanted e-mail, which Internet users commonly call spam.

III. LIMITATIONS OF E-MAIL SECURITY PROTOCOLS

Secure Socket Layer (SSL) [11] and Secure SMTP over TLS [12] are encryption based methods that respectively create encrypted secure channel between the sending and receiving MTA's at sockets and transport layers. They are simple methods to obtain e-mail privacy without efforts of the end user but Secure SMTP over TLS guards only the path between client and server and not the endpoints that are authenticated by certifying authorities and not the Domain Name System (DNS) [13]. Cryptography based encryption techniques for e-mail security includes Privacy-Enhanced Mail (PEM) [14], Pretty Good Privacy (PGP) [15], GNU Privacy Guard (GPG) [16] and Secure Multi-purpose Internet Mail Extensions (S/MIME) [17, 18]. PEM lacks flexibility and more seriously requires trusting a single Certificate Authority (CA) infrastructure which is the reason for its almost negligible adoption [19]. PGP and GPG are PKI based scheme with sporadic adoption and as such are limited to a smaller user community.

.

3.1 SMTP vulnerabilities [28]

There is a big vulnerability that is present in the SMTP protocol which allows once logged in users to send illegitimate emails. The connection to a SMTP server is established using the telnet command

> telnet smtp.exampe.com 25

This command opens a connection to the server providing email server at port number 25.Usually the response is of the

form: 220 smtpmailserver.ontheinternet.com Microsoft ESMTP MAIL Service, Version: 5.0.2195.6713 ready at Mon, 11 Apr 2005 11:15:50 -0400

This means you are successfully connected to the Server! Next we issue a command to say hello to the gateway.

> "hello"

Response:"250smtpmailserver.ontheinternet.com [10.1.1 .x]" This means that the gateway greets you!

Then the rcpt to command is issued to specify the Recipient.

> "rcpt to: person@targetdomain.com":

Who are we sending the e - mail to?

Response: "250 2.1.5 person@targetdomain.com"

This means that we are close to sending our spoofed e-mail message!!!

Action: "data (then hit enter)":

Tell the smtp server we are writing our message next!

Response: "354 Start mail input; end with <CRLF>.<CRLF>":

The mail server is telling us to write our message then type "enter" a period "." , then "enter" again

Result: You type your message

Action: "(Hit enter) type "." (Hit enter)":

Tell the smtp server we are finished writing our message!
Response: "2502.6.0<smtpmailserverWQm21OesnsI0000148e@smtpmailserver.ontheinternet.com>Queued mail for delivery"

Result: The SMTP mail server has just accepted your e-mail for delivery and has queued it for sending!
It is evident from the above example that the mail server does not authenticate the sender email ID and it may be duplicated. Sending fake emails from SMTP server is therefore a common practice.

3.2 Vulnerabilities in Email Security

Email security must consider three fundamental dimensions of vulnerability:

- **Inbound email threats** such as spam, phishing attacks, malware, spyware, blended threats, scams, and spoofs.
- **Outbound vulnerabilities** and liabilities including accidental data loss, intentional data leakage, botnet activity, and contaminated outbound mail. Outbound protection strategies must protect email in transit.
- **Risks associated** with email within the organization including inappropriate sharing of sensitive data and malware contamination. Organizations must protect email "at rest" – both the contents of user inboxes and folders as well as email archives.

IV. PROBLEM STATEMENT

Email is send by the SMTP protocol. Due to the limitation of the SMTP protocol any user can connects to the SMTP server and can compose an email with the fake header information. Because of the limitations SMTP server accept the email without any verification of source of the email and user can be a victim of the spoofing attack.

V. TRUST MANAGEMENT SYSTEM

Systems in which multiple entities share resources often use an access control mechanism. The problem of access control can be broken into two sub problems: determining whether or not a request should be allowed, and enforcing the decision. Trust management systems solve the first sub problem by defining languages for expressing authorizations and access control policies, and by providing a trust management engine for determining when a particular request is authorized. Traditional access control mechanisms are centralized and operate under a closed world assumption in which all of the parties are known. Trust management systems generalize traditional mechanisms by operating in distributed systems and eliminating the closed world assumption. Over the last ten years, a number of trust management systems have been developed, some focusing on authentication [20, 21, 22], others for specialized purposes [3, 8, 18], others for general purpose authorization [4, 6, 13], and others based on logics [1, 2, 17]. Because of the wide range in precision in the specification of these systems and the wide variety of trust management languages, it is difficult to compare the systems in order to intelligently decide which to apply to a new situation. Because of the lack of formality in many of the specifications, it is difficult to understand their weaknesses, which is especially troubling since the domain of interest is often security related. Finally, because there is no common conceptual framework underlying the systems, it is difficult to reason about the trade-offs made in their design. This makes the design of new trust management systems more of an art than a science

Trust and reputation, a closely related term, are firmly rooted in sociology, and those roots should not be forgotten. However, trust is quite a complicated phenomenon, the concept itself carrying many meanings

Trust is defined as the extent to which one party is willing to participate in given action with a given partner considering the risk. Here the trust decision is binary based on the balance of the trust and risk which is posed by the trustee.

The task of the trust management is

- Initializing a trust relationship
- Observation
- Evolving reputation and trust

A TM language has a mechanism for identifying principals, a syntax for specifying policy statements and queries, and a semantic relation that determines whether a query is true given a set of policy statements.

5.1 Deduction

Deduction implements the semantic of the language. A TM system may have the following deduction engines (algorithms). A proof checking engine takes a set of policy statements, a query, and an answer as input, and verifies that the answer is true. The answer may be equipped with proofs (or proof hints) to make proof checking simpler. A proof construction engine (also known as a chain discovery engine) takes a set of policy statements and a query as input, and finds an answer, optionally constructing proofs (or proof hints). In systems that may have a large number of (e.g., millions of) policy statements stored in a de-centralized manner, a chain discovery engine does not have the complete set of policy statements as input, and should be able to start evaluation with a query and an incomplete set of policy statements and to interleave retrieving policy statements and inference.

ATM system can provide a more expressive global-scale public-key infrastructure, by having digital credentials for driver licenses, student IDs, credit cards, organization memberships, trusting relationships regarding these digital credentials, and so on. In these cases, online transaction may require the combination of these credentials.

VI. EVALUATING E-MAIL SERVERS

Availability of free e-mail accounts with or without POP3 and IMAP access through some commercial e-mail service providers has increased the popularity of this very Internet application. However, this has also increased the security risks as spammers and hackers try to reach more and more people through this application for their illicit financial gains. Several anti-spoofing standards like Sender ID/SPF and DKIM successfully validate sending domains. They are not, however, strictly being used in all e-mail servers. Spoofed e-mails from domains that do not follow any standardized anti-spoofing standard are not detected by receiving e-mail servers.

Now in email the sender identities is very important aspects for Email security and the sender related trust is very important in the trust management system the sender's identity is verified by the receiver email server's trust management policies. The receiver email server only accepts the encrypted email with the public key encryption technique and the digital signature for the trust relation between the sender and receiver for email communication.

VII. DIGITAL SIGNATURE TECHNOLOGY

7.1 Functions of Digital Signature

Digital Signature is a method to encrypt a message (such as documents, contracts, notifications) which will be transferred, adopting data-exchanging protocol and data-encrypting algorithm. An abstract is produced in this procession. The abstract is like signature or seal which can be used by receiver to verify the identity of the sender [4]. The functions of digital signature: (1) Assuring data integrity. Once the message changes a little, the abstract will change a lot for hash function's peculiarity, so that avoids the message being distorted. (2)Anti-deniability. Using public key cryptography algorithm, the sender can't deny that he has sent the message for he has the private key. (3)Avoiding receivers forging message that is claimed to be from the sender.

7.2 Public Key Encrypting Scheme

As the base of digital signature technology, public key encrypting technology should be introduced first in the following content. In the traditional cryptography system, the cipher code

Used in the process of encrypting plain text into cipher text and in the inverse process is the same. This method is called symmetric cryptography technology. Public key encrypting scheme is a kind of a symmetric cryptography technology. It resolves the difficult problems in application. Its basic idea: the keys of the two parties are different. Every user has a key pair. One is private key which is saved by the user himself, another one is issued in public places such as internet for downloading or enquiring. Public key algorithm is very slow (with contrast to private algorithm). It is designed for a little data, but not for much data. It is usually used together with hash function in digital signature.

7.3 Functions of Digital Signature

Hash algorithm is an algorithm which is used to compute a data fingerprint of a data block. It is a one-way function which satisfies the following conditions:

- Can receive data with any length;
- Can produce abstract with fixed length;
- Can compute abstract easily;
- Cannot compute message from abstract;
- It is impossible to find two different messages which have same abstract. Hash function can make short abstract with fixed length for the binary data with any length. The popular hash algorithms are MD5, Secure Hash Algorithm (SHA, having all kinds of security level.)

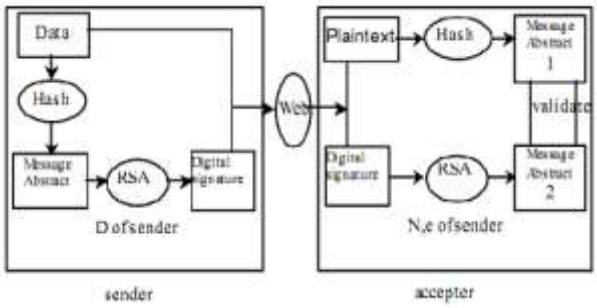

Fig. 1 A Digital Signature Algorithm

VIII. PUBLIC KEY ENCRYPTION FOR EMAIL

The primary benefit of public key cryptography is that it allows people who have no pre-existing security arrangement to exchange messages securely. The need for sender and receiver to share secret keys via some secure channel is eliminated; all communications involve only public keys, and no private key is ever transmitted or shared. [19]. to send an encrypted mail to somebody, you encrypt it using the public key. Only the addressee himself will be able to decrypt it using his private key.

Public key encryption is only safe and secure if the sender of an enciphered message can be sure that the public key used for encryption belongs to the recipient. A third party can produce a public key with the recipient's name and give it to the sender, who uses the key to send important information in encrypted form. The enciphered message is intercepted by the third party, and since it was produced using their public key they have no problem deciphering it with their private key.

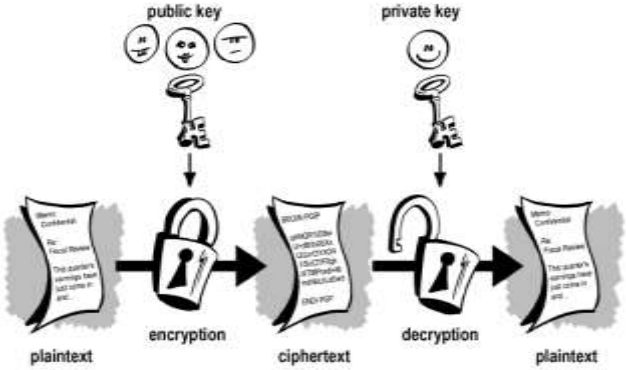

Fig. 2 Email Encryption Using Public key Cryptosystem algorithm

(http://www.akadia.com/services/email_security.html)

IX. EMAIL BASED TRUST MANAGEMENTAPPROCH AND FUTURE

In the email based trust management system email is send by sender to receiver by the SMTP server here by the SMTP server never check the authentication for sender and at the receiver end it will not verify that email is send by the intended user for this reason email is spoof by the attacker by using spoof identity and send by any other SMTP server connect by the telnet and at the receiver end it will not verified that email is send by the authenticated server. As Shown in the figure client1 send the email using SMTP server and at the other end client2 accept email by their SMTP server

Fig. 3 Email Working System

As shown in the Fig 4. Client A Send the email using SMTP server and that email is encrypted [19,20,21] by public key of the receiver which is maintain by the central trusted authority this authority is trusted based on the role based trust management system that authority is trusted by the certified Agencies. Now email is secure and only be decrypted by the receivers private key but if there is any problem with the security at the receiver end the SMTP server of the receiver check the digital signature for the communication if it is not valid that email is discard by the server here all system is implemented based on the trusted authority and trust management system

In the propose method email is fist digitally sign and then encrypt it with the public key encryption system with the centralize KDC (key distribution Centre) it contain two pair of key public and private key. Email is email is encrypted with the receiver's public key so that only receiver can decrypt it.

At the receiver end it will verified the digital signature if the email is not digital sign then it will be ignore and removed if digital signature is verified then it will going for decryption phase and finally receiver can view that email is verify and digitally sign and verified by trusted authority by using role based trust management system.

9.1 Email Transfer Step by Source Server using proposed method

- Email Send by the user to SMTP server
- SMTP server encrypt the email with the receiver's public key
- Digital Certificate is attached with the email
- Email is Transfer from Source SMTP server to the Receiver's SMTP server

9.2 Email Capture step by the receiver using propose method

- Receiver Server Catch the email from the source
- Verify for the digital Certificate
- Decrypt the email with the private key of the receiver
- Forwarded to the Receiver

Fig. 4 Email System Using Trust management System

X. CONCLUSION

We have proposed method for email security by using the public key encryption with digital sign signature so that the email authenticity will maintain both the side. Sender sends encrypted email using public key encryption algorithm and then the email is certified by certificate authority so that the receiver server will accept that email only which is digitally sign and only decrypted with the receiver's private key. The digital signature is maintain by the central authority and public and private key is maintain by KDC so and only authorize server get the public key for encryption and decryption.

REFERENCES

[1] Klensin, (2001) 'Simple Mail Transfer Protocol' IETF RFC 2821.
[2] Die, W., and Hellman, M. Exhaustive cryptanalysis of the NBS data encryption Standard. Computer 10(June1977), 74-84.
[3] Knuth, D.E. The Art of Computer Programming, Vol2:SeminumericalAlg——orithms.Addison-Wesley, Reading, Mass., 1969.
[4] 4 Malkin T, Micciancio D, Miner S. Effcient Generic Forward-secure Signatures with an Unbounded Number of Time Periods[C]. Proc. Of Advances in Cryptology-EUROCRYPT. 2002.
[5] Merkle, R. Secure communications over an in secure channel. Submitted to Comm. ACM.
[6] Miller, G.L. Riemann's hypothesis and tests for primarily. Proc. Seventh Annual ACMSymp. On the Theory of Computing. Albuquerque, NewMex. May1975, pp. 234-239; extendedvers.availableasRes.Rep.CS-75-27, Dept. of Computer .Sci. U.of Waterloo, Waterloo, Ont., Canada, Oct.1975.
[7] Niven, I., and Zuckerman, H.S. An Introduction to the Theory of Numbers. Wiley, New York, 1972
[8] S. T. Kent, (1993) "Internet Privacy Enhanced Mail" Communications of ACM, Vol. 36, No. 8, pp. 40-42.
[9] Pollard, J.M.Theorems on factorization and primarily testing. Proc.Camb.Phil. Soc.76 (1974), 521-528.
[10] Potter, R.J., Electronic mail. Science195, 4283(March1977), 1160-1164.
[11] Tahir Elgamel, and Kipp E. B. Hipman, (1997) "Secure Socket Layer Application Program Apparatus and Method" U.S. Patent No: 5657390.
[12] P. Hoffman, (2002) "SMTP Service Extension for Secure SMTP over Transport Layer Security", IETF RFC 3207.
[13] S. Suzuki and M. Nakamura, (2005) "Domain Name System—Past, Present and Future", IEICE Transactions of Communication, E88b (3), pp. 857-864.
[14] S. T. Kent, (1993) "Internet Privacy Enhanced Mail" Communications of ACM, Vol. 36, No. 8, pp. 48-60.
[15] PGP, (ND) "Pretty Good privacy (PGP)", http://www.pgp.com, accessed 25 August, 2009.
[16] W. Koch, (ND) "The GNU privacy guard", http://www.gnupg.org, accessed 7 September 2009.
[17] B. Ramesdell, (2004) "Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 3.1 Message specification", Internet Engineering Task Force (IETF), RFC 3851.

[18]] B. Ramesdell, (2004) "Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 3.1 Certificate Handling", Internet Engineering Task Force (IETF), RFC 3850.
[19] http://www.akadia.com/services/email_security.html
[20] http://en.wikipedia.org/wiki/Pretty_Good_Privacy
[21] http://www.instantcrypt.com/
[22] http://www.ccs.neu.edu/home/riccardo/courses/csg399-sp06/trust-management.pdf
[23] http://spdp.dti.unimi.it/papers/tweb2012.pdf
[24] www.doc.ic.ac.uk/~tgrand/iTrust.pdf
[25] http://www.tcpipguide.com/free/t_SMTPSecurityIssues.htm
[26] Kunal pandove,Amandip Jindal and Rajnidar Kumar , "Email Spoofing" in Volume- 5 No1 International Journal of computer Application August 2010.
[27] Kunal pandove,Amandip Jindal and Rajnidar Kumar , "Email Security" in Volume- 5 No1 International Journal of computer Application August 2010.
[28] Kunal pandove,Amandip Jindal and Rajnidar Kumar , "Launching Email Spoofing Attacks" in Volume- 5 No1 International Journal of computer Application August 2010