TECHNIQUES AND TOOLS FOR FORENSIC INVESTIGATION OF E-MAIL

M. Tariq Banday

P. G. Department of Electronics and Instrumentation Technology
University of Kashmir, Srinagar - 6, India
sgrmtb@yahoo.com

ABSTRACT

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 e-mails 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.

KEYWORDS

E-mail Forensics; E-mail Headers; E-mail Security; Header Analysis; E-mail Architecture

1. INTRODUCTION

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 figure 1.

'*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 [1] '*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* [2] or *IMAP* [3] 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.

DOI : 10.5121/ijnsa.2011.3617

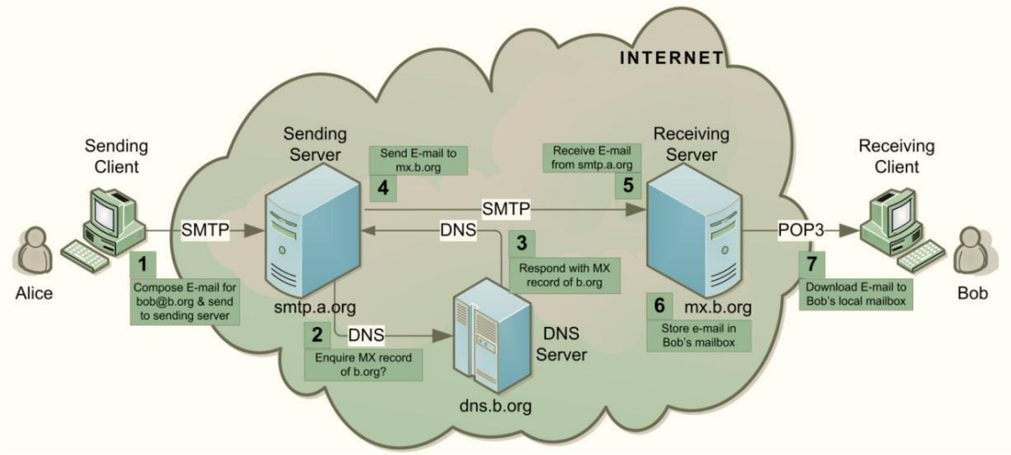
INTERNET
Sending Client
Alice
SMTP
1
Compose E-mail for bob@b.org & send to sending server
Sending Server
smtp.a.org
Send E-mail to mx.b.org
4
2
DNS
Enquire MX record of b.org?
DNS
3
Respond with MX record of b.org
DNS Server
dns.b.org
SMTP
Receive E-mail from smtp.a.org
5
Receiving Server
6
mx.b.org
Store e-mail in Bob's mailbox
POP3
7
Download E-mail to Bob's local mailbox
Receiving Client
Bob

Figure 1: *E-mail communication between a sender 'Alice' and recipient 'Bob'*

2. E-MAIL ACTORS, ROLES AND RESPONSIBILITIES

E-mail is a highly distributed service involving several actors that play different roles to accomplish end-to-end mail exchange [4]. These actors fall under "User Actors", "Message Handling Service (*MHS*) Actors" and "ADministrative Management Domain (*ADMD*) Actors" groups.

User Actors are people, organizations or processes that serve as sources or sinks of messages. They can generate, modify or look at the whole message. User Actors can be of following four types (Table 1):

User Actor Type	Roles and Responsibilities
Author	▪ Responsible for creating the message, its contents, and its list of Recipient addresses. ▪ The MHS transfers the message from the Author and delivers it to the Recipients. ▪ The MHS has an Originator role that correlates with the Author role.
Recipient	▪ The Recipient is a consumer of the delivered message. ▪ The MHS has a Receiver role that correlates with the Recipient role. ▪ A Recipient can close the user-communication loop by creating and submitting a new message that replies to the Author e.g. an automated form of reply is the Message Disposition Notification (MDN)
Return Handler	▪ It is a special form of Recipient that provides notifications (failures or completions) generated by the MHS as it transfers or delivers the message. ▪ It is also called Bounce Handler.
Mediator	▪ It receives, aggregates, reformulates, and redistributes messages among Authors and Recipients. ▪ It forwards a message through a re-posting process. ▪ It shares some functionality with basic MTA relaying, but has greater flexibility in both addressing and content than is available to MTAs. It preserves the integrity and tone of the original message, including the essential aspects of its origination information. It might also add commentary. ▪ It does not create new message that forwards an existing message, Reply or annotation. ▪ Some examples of mediators are: Alias, ReSender, Mailing Lists, Gateways and Boundary Filter.

Table 1: User Actors and their Responsibilities

All types of Mediator user actors set *HELO*/*EHLO*, *ENVID*, *RcptTo* and *Received* fields. Alias actors also typically change *To*/*CC*/*BCC* and *MailFrom* fields. Identities relevant to *ReSender* are: *From*, *Reply-To*, *Sender*, *To*/*CC*/*BCC*, *Resent-From*, *Resent-Sender*, *Resent-To*/*CC*/*BCC* and *MailFrom* fields. Identities relevant to Mailing List processor are: *List-Id*, *List-**, *From*, *Reply-To*, *Sender*, *To*/*CC* and *MailFrom* fields. Identities relevant to Gateways are: *From*, *Reply-To*, *Sender*, *To*/*CC*/*BCC* and *MailFrom* fileds.

Message Handling Service (MHS) Actors are responsible for end-to-end transfer of messages. These Actors can generate, modify or look at only transfer data in the message. *MHS* Actors can be of following four types (Table 2):

MHS Actor Type	Roles and Responsibilities
Originator	▪ It ensures that a message is valid for posting and then submits it to a Relay ▪ It is responsible for the functions of the Mail Submission Agent. ▪ It also performs any post-submission that pertain to sending error and delivery notice. ▪ The Author creates the message, but the Originator handles any transmission issues with it
Relay	▪ It performs MHS-level transfer-service routing and store-and-forward function by transmitting or retransmitting the message to its Recipients. ▪ It adds trace information but does not modify the envelope information or the semantics of message content. ▪ It can modify message content representation, such as changing the form of transfer encoding from binary to text, but only (as required) to meet the capabilities of the next hop in the MHS. ▪ When a Relay stops attempting to transfer a message, it becomes an Author because it sends an error message to the Return Address.
Gateway	▪ It connects heterogeneous mail services despite differences in their syntax and semantics. ▪ It can send a useful message to a Recipient on the other side, without requiring changes to any components in the Author's or Recipient's mail services.
Receiver	▪ It performs final delivery or sends the message to an alternate address. ▪ It can also perform filtering and other policy enforcement immediately before or after delivery.

Table 2: MHS Actors and their Responsibilities

ADministrative Management Domain (ADMD) Actors are associated with different organizations which have their own administrative authority, operating policies and trust-based decision making. *ADMD* Actors can be of following three types (Table 3):

User Actor Type	Roles and Responsibilities
Edge	▪ Independent transfer services in networks at the edge of the open Internet Mail service.
Consumer	▪ Might be a type of Edge service, as is common for web-based email access.
Transit	▪ E-Mail Service Providers (ESPs) that offer value-added capabilities for Edge ADMDs, such as aggregation and filtering.

Table 3: ADMD and their Responsibilities

The mail-level transit service is different from packet-level switching. End-to-end packet transfers usually go through intermediate routers; e-mail exchange across the open Internet can be directly between the Boundary *MTAs* of Edge *ADMDs*. Edge networks can use proprietary email standards internally. Common examples of *ADMDs* are: Enterprise Service Providers, Internet Service Providers (*ISP*) and E-mail Service Providers.

3. E-MAIL ARCHITECTURE

E-mail system is an integration of several hardware and software components, services and protocols which provide interoperability between its users and among the components along the path of transfer. The e-mail architecture shown in figure 2 below specifies the relationship between its logical components for creation, submission, transmission, delivery and reading processes of an e-mail message.

Several communicating entities called e-mail nodes which are essentially software units working on application layer of *TCP/IP* model are involved in the process of e-mail delivery. Nodes working on lower layers such as routers and bridges represent options to send e-mail without using *SMTP* are not considered in this architecture because almost all e-mail communication uses *SMTP* directly or indirectly. Further, proprietary nodes used for internal deliveries at sending and receiving servers are also not considered in this architecture.

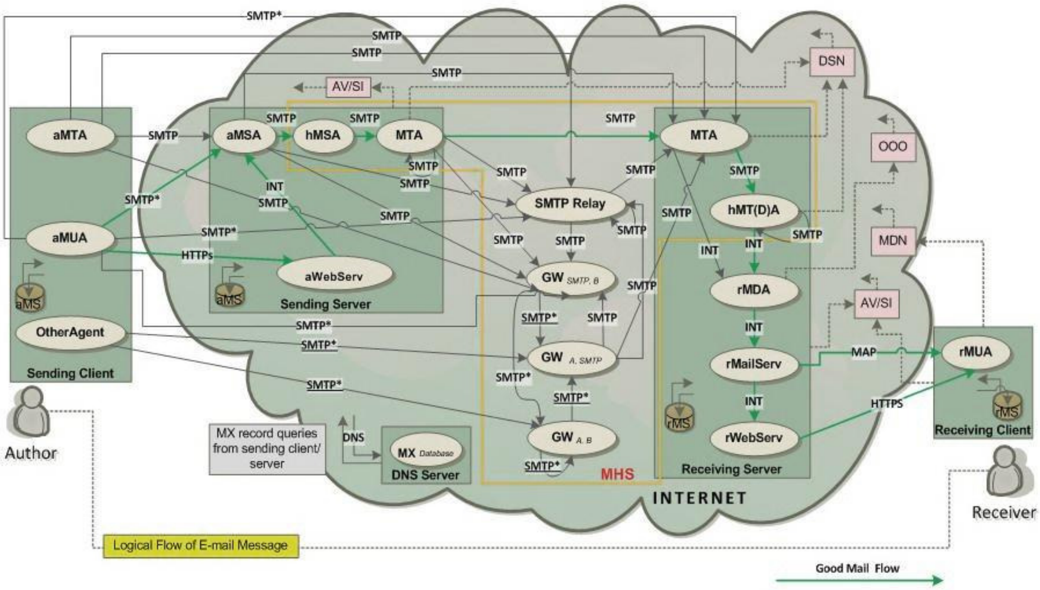

Figure 2: *E-mail Architecture showing relationship between its various components*

A mail message from *Author* to *Receiver* that traverses through *aMUA, aMSA, hMSA, MTA (outbound), MTA (Inbound), hMDA, rMDA, rMailServ* and *rMUA* is considered as good mail by the Sender Policy Forum (*SPF)*. Mails following through other paths are either fully or partially non-SMTP based or uses non-standard transfer modes which are often suspected to contain viruses and spam. Delivery Status Notification (*DSN*) messages are generated by some components of *MHS* (*MSA*, *MTA*, or *MDA*) which provide information about transfer errors or successful deliveries and are sent to *MailFrom* addresses. Message Disposition Notification (*MDN*) messages are generated by *rMUA* which provide information about post-delivery processing are sent to Disposition-Notification-To address. Out Of Office (*OOO*) messages are sent by *rMDA* to return address. The functions and roles of various components shown in the architecture are discussed below.

Message/Mail User Agent (MUA): It works for user actors and applications as their representative within e-mail service. A *MUA* that works on behalf of Author is called Author *MUA* (*aMUA*) and the one that works on behalf of Receiver is called Receiver *MUA* (*rMUA*). *aMUA* creates messages and performs initial submission via Mail Submission Agent (*MSA*). Besides this, it can also perform creation and posting time archiving in its Message Store. *rMUA* processes received mail that includes generation of user level disposition control messages, displaying and disposing of the received message and closing or expanding the user communication loop by initiating replies and forwarding new messages. A Mediator performs message re-posting and as such it is a special *MUA*. For bulk sending services and automatic responder (serving out of office notifications), *MUA* can be automated. The identity fields relevant to MUA are: *From*, *Reply-To*, *Sender*, *To*, *CC* and *BCC*.

All Mail User Agent (*MUA*) nodes are software packages that run on client computers and allow end users to compose, create and read e-mail. Some *MUAs* may be used to send e-mail to the receiving *MTAs* directly or indirectly. '*Microsoft Outlook*', '*Microsoft Outlook Express*', '*Lotus Notes*', '*Netscape communicator*', '*Qualcomm Eudora*', '*KDE KMail*', '*Apple Mail*', and '*Mozilla Thunderbird*' are examples of *MUAs*. Several Web-based e-mail programs and services (known as Webmail) such as '*AIM Mail*', '*Yahoo Mail*', '*Gmail*', and '*Hotmail*' which integrate e-mail clients and servers behind a Web server are also used as *MUAs*.

Message/Mail Store (MS): It serves as a long term message store for *MUA* which can be located on a remote server or on the machine running *MUA*. Messages can be organized in a *MS* in different ways. The *MUA* accesses the *MS* either by a local mechanism or by using *POP* or *IMAP*.

Message/Mail Submission Agent (MSA): Mail Submission Agent (*MSA*) accepts the message submitted by the *aMUA* for posting. It enforces the policies of the hosting *ADMD* and the requirements of Internet standards before posting the message from an Authors environment to the *MHS*. These include adding header fields such as *Date* and *Message-ID* and expanding an address to its formal Internet Mail Format (*IMF)* representation. The *hMSA* is responsible for transiting the message to *MTA*. The identity fields relevant to *MSA* are: *HELO/EHLO*, *ENVID*, *MailFrom*, *RcptTo*, *Received*, and *SourceAddr*. The responsibilities of *MUA* and *MSA* may be integrated in a single Agent.

Message/Mail Transfer Agent (MTA): A Message Transfer Agent (*MTA*) relays mail for one application-level "hop". *MTA* nodes are in effect postal sorting agents that have the responsibility of retrieving the relevant Mail eXchange (***MX***) record from the *DNS Server* for each e-mail to be send and thus map the distinct e-mail addressee's domain name with the relevant IP address information. *DNS* is a distributed directory database that correlated domain names to IP addresses. *MTAs* can also be used to compose and create e-mail messages. '*Sendmail*', '*Postfix*', '*Exim*', and '*Exchange Server*', are examples of *MTAs*. A receiving *MTA* can also perform the operation of delivering e-mail message to the respective mailbox of the receiver on the mail server and thus is also called Mail Delivery Agent (*MDA*). Unlike typical packet switches (and Instant Messaging services), *MTAs* are expected to store messages in a manner that allows recovery across service interruptions, such as host-system shutdown. The offered degree of robustness and persistence by *MTAs* can vary. An *MTA* can perform well established roles of Boundary *MTAs* (*Onbound* or *Inbound*) or Final *MTAs*. The identity fields relevant to MTAs are: *HELO/EHLO*, *ENVID*, *MailFrom*, *RcptTo*, *Received*, and *SourceAddr.*

Message/Mail Delivery Agent (MDA): Both *hMDA* and *rMDA* are responsible for accepting the message for delivery to distinct addresses. *hMDA* functions as a SMTP server engine and *rMDA* performs the delivery action. The identity fields relevant to *MDA* are: *Return-Path* and *Received*.

Relays: *SMTP-Relays* are the nodes that perform e-mail relaying. Relaying is the process of receiving e-mail message from one SMTP e-mail node and forward it to another one. They are like packet switches or IP routers and make routing assessments to move the message closer to the Recipients. They also add trace information and have all roles of *MTA's*.

Gateway: Gateway nodes are used to convert e-mail messages from one application layer protocol to other. Gateway nodes named $GW_{SMTP,\ B}$ accept SMTP protocol based e-mails and transfer them with protocols other that SMTP and $GW_{A,\ SMTP}$ performs the inverse process at incoming and outgoing interfaces. Gateway nodes $GW_{A,B}$ do not use SMTP either for incoming or outgoing interfaces. A process called Proxy may be done at these nodes when incoming and outgoing interfaces use same protocols.

Web Server (WebServ): These nodes are the e-mail Web servers that provide the Web environment to compose, send and read an e-mail message.

Mai Server (MailServ): They represent e-mail servers providing users mail access service using *IMAP* or *POP3* protocols. They can also provide an internal interface to a Web server for *HTTP* based e-mail access.

The e-mail nodes establish connections with one or more nodes on specific ports for possible e-mail flow between them using a particular protocol. *SMTP* is an application layer protocol for *TCP/IP* based Internet infrastructure which sets conversational and grammatical rules for exchanging e-mail between computers. The most commonly-used protocols for e-mail retrieval by client programs are Post Office Protocol Version 3 (*POP3*) and Internet Message Access Protocol (*IMAP*). Table 4, lists the protocols used in e-mail flow between two possible e-mail nodes.

Protocol Group	**Protocols**
$SMTP$	SMTP protocol (RFC 821), SMTP service extension protocols ESMTP including Service Extension for Authentication (RFC 2554), Delivery by SMTP Service Extension (RFC 2852), SMTP Service Extension for Routing Enhanced error (RFC 2034) and SMTP Service Extension for Secure SMTP over Transport Layer Security (RFC 3207).
$SMTP^{-}$	All protocols in SMTP group and all SMTP extensions for e-mail submission from MUA to e-mail node with SMTP incoming interface. E-mail node can be MTA defined in RFC 2821, MSA defined in RFC 2476. Using MSA various methods can be applied for ensuring authenticating user that include IP address restrictions, secure IP and POP authentication.
*SMTP**	All Internet application protocols except those specified in $SMTP^{-}$ group, all preparatory application protocols used on the Internet (also used for tunneling), all Internet protocols on the transport and network layers such as TCP/IP as it is possible to send e-mail without the use of application layer protocols.
$HTTP(S)$	HTTP (RFC 2616), HTTP over SSL and HTTP over TLS (RFC 2818).
INT	ESP specific protocols and procedures for internal e-mail delivery between e-mail nodes.

MAP	All e-mail access protocols used to transfer e-mails from the recipient e-mail server to MUA that include IMAP version 4 (RFC 1730), MAPI and POP version 3 (RFC 1939).

Table 4: Protocols used in E-mail Transfer

For networks, a port means an endpoint to a logical connection. The port number identifies what type (application/service offered) of port it is. The commonly used default port numbers used in e-mail are shown in Table 5. A complete list of default port numbering assignment is given in [5].

Port No	Protocols/Services	Description
25	SMTP SMTP e-mail server	Simple Mail Transfer Protocol - core Internet protocol used to transfer from client to server (MUA to MTA) and server to server (MTA to MTA)
110	POP3 POP e-mail server	Post Office Protocol allows clients (MUA's) to retrieve stored e-mail
143	IMAP IMAP(4) e-mail server	Internet Message Access Protocol provides a means of managing e-mail messages on a remote server and retrieve stored e-mail
465	SMTPS WSMTP (SSMTP) protocol over TLS|SSL	SMTP via SSL encrypted connection (Unofficial)
993	IMAPS SSL encrypted IMAP	IMAP via SSL encrypted connection
995	POP3S SPOP SSL encrypted POP	POP via SSL encrypted connection
587	MSA	Outgoing Mail (Submission)
80	HTTP	Webmail
443	HTTPS	Secure Webmail

Table 5: Commonly used Ports in E-mail Communication

4. E-MAIL IDENTITIES AND DATA

Identities used in E-mail are globally unique and are: *mailbox*, *domain name*, *message-ID* and *ENVID*. *Mailboxes* are conceptual entitities identified by e-mail address and receive mail. E-mail address has become a common identity identifier on the Internet. An e-mail address consists of username and domain name separated by @ sign e.g. *aliace@a.com*. Ray Tomlinson first initiated the use of @ sign to separate username from the domain name. A domain name is a global reference to an Internet resource like a host, network or service which maps to *IP* address(es). Its structure has a hierarchical sequence of labels, separated by dots. The top of the hierarchy is on the right end of the sequence e.g. *mail.kashmiruniversity.ac.in*. *Message-ID* and

ENVID are message identifiers which respectively pertain to message content and transfer. *Message-ID* is used for threading, aiding identification for duplications and *DNS* tracking. The ENVelope Identifier (*ENVID)* is used for the purpose of message tracking. *Message-ID* and *ENVID* are discussed further in the section 7.

E-mail message comprises of envelope that contains transit-handling information used by the *MHS* and message content which consists of two parts namely Body and Header. The Body is text but can also include multimedia elements in Hyper Text Markup Language (*HTML*) and attachments encoded in Multi-Purpose Internet Mail Extensions (*MIME*) [6]. The Header is a structured set of fields that include '*From'*, '*To'*, '*Subject'*, '*Date'*, '*CC'*, '*BCC'*, '*Return-To'*, etc. Headers are included in the message by the sender or by a component of the e-mail system and also contain transit-handling trace information. Further, the message also contains special control data pertaining to Delivery Status and Message Disposition Notifications, etc.

Various identities called fields are present in the message and are used in different parts of e-mail architecture called Layers. These fields serve a specific function in the system and are set by some component of the system. Table 6 lists main identifier fields present in the message during transit along with their description and the actor responsible for specifying their value. These identities are used for analysing e-mail to determine the source (originator and the author).

Field Name	Set By	Field Description
Layer: ***Message Header Fields (Identification Fields)***		
Message- ID:	Originator	Globally unique message identification string generated when it is sent.
In-Reply-To:	Originator	Contains the Message-ID of the original message in response to which the reply message is sent.
References:	Originator	Identifies other documents related to this message, such as other e-mail message.
Layer: ***Message Header Fields (Originator Fields)***		
From:	Author	Name and e-mail address of the author of the message
Sender:	Originator	Contains the address responsible for sending the message on behalf of Author, if not omitted or same as that specified in From field.
Reply- To:	Author	E-mail address, the author would like recipients to use for replies. If present it overrides the From field.
Layer: ***Message Header Fields (Originator Date Fields)***		
Date:	Originator	It holds date and time when the message was made available for delivery.
Layer: ***Message Header Fields (Informational Fields)***		
Subject:	Author	It describes the subject or topic of the message.
Comments:	Author	It contains summarized comments regarding the message.
Keyword:	Author	It contains list of comma separated keywords that may be useful to the recipients e.g. when searching mail.
Layer: ***Message Header Fields (Destination Address Fields)***		
TO:	Author	Specifies a list of addresses of the recipients of the message. These addresses might be different from address in RcptTo SMTP commands
CC:	Author	Generally same as To Field. Generally a To field specifies primary recipient who is expected to take some action and CC addresses

Field Name	Set By	Field Description
		receive a copy as a courtesy.
BCC:	Author	Address of recipient whose participation is not disclosed to recipients specified in To and CC addresses.
Layer: ***Message Header Fields (Resent Fields)***		
Resent-Message- ID:	Mediator	Globally unique message identification string generated when it is resent.
Resent-*	Mediator	When manually forwarding a message, resent header fields referring to the forwarding, not to the original message. MIME specifies another way of resending messages, using the "Message" Content-Type.
Layer: ***Message Header Fields (List Fields)***		
List-ID	Mediator, Author	Globally unique Mailing List identification string.
List-*	Mediator, Author	A collection of header fields for use by Mailing Lists.
Layer: ***Message Header Fields (Trace Fields)***		
Received:	Originator, Relay, Mediator, Destination	Contains trace information that includes originating host, Mediators, relays, and MSA host domain names and/or IP addresses
Return-Path:	MDA, from MailFrom	Contains the address recorded by MDA from MailFrom SMTP command
Layer: ***Message Header Fields (Optional Security Fields)***		
DKIM Signature	MUA, MSA or MTA	The signature of the email is stored in the DKIM-Signature header field. This header field contains all of the signature and key-fetching data. DKIM uses a simple "tag=value" syntax in several contexts, including in messages and domain signature records
Received-SPF	MTA	It contains Sender Policy Framework (SPF) validation results for a domain and its mail servers. Domain owners publish records via DNS that describe their policy for which machines are authorized to use their domain in the HELO and MAIL FROM addresses, which are part of the SMTP protocol.
Layer: ***Message Body Fields (MIME Header Fields)***		
MIME-Version	Author	It describes the version of the MIME message format.
Content-*	Author	It contains a collection of MIME Header fields describing various aspects of message body, including and signatures.
Layer: ***SMTP***		
HELO/EHLO	Latest Relay Client (Originator, MSA, MTA)	It contains the hosting domain for the SMTP HELO and EHLO commands.
ENVID	Originator	An opaque string included in DSN as a means for assisting the Return Address Recipient in identifying the message that produced a DSN or message tracking.

Field Name	Set By	Field Description
MailFrom	Originator	It is a string containing e-mail address for receiving return control information like returned messages transfer level problems)
RcptTo	Author	It specifies MUA mailbox address of a recipient.
ORCPT	Originator	Is an optional parameter to the RCPT command, indicating the original address to which the current RCPT TO address corresponds after a mapping during transit.
Layer: ***IP***		
Source Address	Latest Relay Client	It contains the source Address of the host immediately preceding the current receiving SMTP server from which the IP datagram (e-mail message is fragmented into IP packets) was send. It is independent of the mail system and is supplied by the IP layer.

Table 6: Main Identifier fields present in a message during transit

5. E-MAIL FORENSIC INVESTIGATION TECHNIQUES

E-mail forensics refers to the study of source and content of e-mail as evidence to identify the actual sender and recipient of a message, data/time of transmission, detailed record of e-mail transaction, intent of the sender, etc. This study involves investigation of metadata, keyword searching, port scanning, etc. for authorship attribution and identification of e-mail scams. Various approaches that are used for e-mail forensic are described in [7] and are briefly defined below:

5.1. Header Analysis

Meta data in the e-mail message in the form of control information i.e. envelope and headers including headers in the message body contain information about the sender and/or the path along which the message has traversed. Some of these may be spoofed to conceal the identity of the sender. A detailed analysis of these headers and their correlation is performed in header analysis.

5.2. Bait Tactics

In bait tactic investigation an e-mail with http: “<img src>” tag having image source at some computer monitored by the investigators is send to the sender of e-mail under investigation containing real (genuine) e-mail address. When the e-mail is opened, a log entry containing the IP address of the recipient (sender of the e-mail under investigation) is recorded on the http server hosting the image and thus sender is tracked. However, if the recipient (sender of the e-mail under investigation) is using a proxy server then IP address of the proxy server is recorded. The log on proxy server can be used to track the sender of the e-mail under investigation. If the proxy server’s log is unavailable due to some reason, then investigators may send the tactic e-mail containing a) *Embedded Java Applet* that runs on receiver’s computer or b) *HTML* page with *Active X Object.* Both aiming to extract IP address of the receiver’s computer and e-mail it to the investigators.

5.3. Server Investigation

In this investigation, copies of delivered e-mails and server logs are investigated to identify source of an e-mail message. E-mails purged from the clients (senders or receivers) whose recovery is impossible may be requested from servers (*Proxy* or *ISP*) as most of them store a copy of all e-mails after their deliveries. Further, logs maintained by servers can be studied to trace the address of the computer responsible for making the e-mail transaction. However, servers store the copies of e-mail and server logs only for some limited periods and some may not co-operate with the investigators. Further, *SMTP* servers which store data like credit card number and other data pertaining to owner of a mailbox can be used to identify person behind an e-mail address.

5.4. Network Device Investigation

In this form of e-mail investigation, logs maintained by the network devices such as routers, firewalls and switches are used to investigate the source of an e-mail message. This form of investigation is complex and is used only when the logs of servers (*Proxy* or *ISP*) are unavailable due to some reason, e.g. when *ISP* or proxy does not maintain a log or lack of co-operation by ISP's or failure to maintain chain of evidence.

5.5. Software Embedded Identifiers

Some information about the creator of e-mail, attached files or documents may be included with the message by the e-mail software used by the sender for composing e-mail. This information may be included in the form of custom headers or in the form of *MIME* content as a Transport Neutral Encapsulation Format (*TNEF*). Investigating the e-mail for these details may reveal some vital information about the senders e-mail preferences and options that could help client side evidence gathering. The investigation can reveal *PST* file names, Windows logon username, *MAC* address, etc. of the client computer used to send e-mail message.

5.6. Sender Mailer Fingerprints

Identification of software handling e-mail at server can be revealed from the *Received* header field and identification of software handling e-mail at client can be ascertained by using different set of headers like "*X-Mailer*" or equivalent. These headers describe applications and their versions used at the clients to send e-mail. This information about the client computer of the sender can be used to help investigators devise an effective plan and thus prove to be very useful.

6. E-MAIL FORENSIC TOOLS

There are many tools which may assist in the study of source and content of e-mail message so that an attack or the malicious intent of the intrusions may be investigated. These tools while providing easy to use browser format, automated reports, and other features, help to identify the origin and destination of the message, trace the path traversed by the message; identify spam and phishing networks, etc. This section introduces some of these tools.

6.1. eMailTrackerPro

eMailTrackerPro [8] analyses the headers of an e-mail to detect the IP address of the machine that sent the message so that the sender can be tracked down. It can trace multiple e-mails at the same time and easily keep track of them. The geographical location of an IP address is key information for determining the threat level or validity of an e-mail message. This tool can pin point the city that the e-mail most likely came from. It identifies the network provider (or *ISP*) of the sender and provide contact information for further investigation. The actual path to the sender's *IP* address is reported in a routing table, providing additional location information to help determine the sender's true location. The abuse reporting feature in it can be used to make further investigation easier. It checks the mail against *DNS* blacklists such as *Spamcop* to further safeguard against spam and malicious emails. It supports Japanese, Russian and Chinese language spam filters besides English language. A major feature of this tool is abuse reporting that can create a report that can be sent to the *ISP* of sender. The *ISP* can then takes steps to prosecuting the account holder and help put a stop to spam.

6.2. EmailTracer

EmailTracer [9] is an Indian effort in cyber forensics by the Resource Centre for Cyber Forensics (RCCF) which is a premier centre for cyber forensics in India. It develops cyber forensic tools based on the requirements of law enforcement agencies. Among several other digital forensic tools, it has developed an e-mail tracer tool named *EmailTracer*. This tool traces the originating *IP* address and other details from e-mail header, generates detailed *HTML* report of email header analysis, finds the city level details of the sender, plots route traced by the mail and display the originating geographic location of the e-mail. Besides these, it has keyword searching facility on e-mail content including attachment for its classification.

6.3. Adcomplain

Adcomplain [10] is a tool for reporting inappropriate commercial e-mail and usenet postings, as well as chain letters and "make money fast" postings. It automatically analyses the message, composes an abuse report, and mails the report to the offender's internet service provider by performing a valid header analysis. The report is displayed for approval prior to mailing to U.S. Federal Trade Commission. Adcomplain can be invoked from the command line or automatically from many news and mail readers.

6.4. Aid4Mail Forensic

Aid4Mail Forensic [11] is e-mail investigation software for forensic analysis, e-discovery, and litigation support. It is an e-mail migration and conversion tool, which supports various mail formats including Outlook (*PST*, *MSG* files), Windows Live Mail, Thunderbird, Eudora, and *mbox*. It can search mail by date, header content, and by message body content. Mail folders and files can be processed even when disconnected (unmounted) from their email client including those stored on CD, DVD, and USB drives. *Aid4Mail Forensic* can search *PST* files and all supported mail formats, by date range and by keywords in the message body or in the headers. Special Boolean operations are supported. It is able to process unpurged (deleted) e-mail from *mbox* files and can restore unpurged e-mail during exportation.

6.5. AbusePipe

AbusePipe [12] analyses abuse complaint e-mails and determines which of ESP’s customers is sending spam based on the information in e-mailed complaints. It automatically generates reports reporting customers violating ESP’s acceptable user policy so that action to shut them down can be taken immediately. *AbusePipe* can be configured to automatically reply to people reporting abuse. It can assist in meeting legal obligations such as reporting on the customers connected to a given IP address at a given date and time.

6.6. AccessData’s FTK

AccessData’s FTK [13] is standard court-validated digital investigations platform computer forensics software delivering computer forensic analysis, decryption and password cracking within an intuitive and customizable interface. It has speed, analytics and enterprise-class scalability. It is known for its intuitive interface, e-mail analysis, customizable data views and stability. It supports popular encryption technologies, such as *Credant, SafeBoot, Utimaco, EFS, PGP, Guardian Edge, Sophos Enterprise* and *S/MIME*. Its current supported e-mail types are: *Lotus Notes NSF, Outlook PST/OST, Exchange EDB, Outlook Express DBX, Eudora, EML* (*Microsoft Internet Mail, Earthlink, Thunderbird, Quickmail, etc.*)*, Netscape, AOL* and *RFC 833*.

6.7. EnCase Forensic

EnCase Forensic [14] is computer forensic application that provides investigators the ability to image a drive and preserve it in a forensic manner using the EnCase evidence file format (*LEF* or *E01*), a digital evidence container vetted by courts worldwide. It contains a full suite of analysis, bookmarking and reporting features. *Guidance Software* and third party vendors provide support for expanded capabilities to ensure that forensic examiners have the most comprehensive set of utilities. Including many other network forensics investigations, it also supports Internet and e-mail investigation. It included Instant Messenger toolkit for *Microsoft Internet Explorer*, *Mozilla Firefox, Opera and Apple Safari*. The e-mail support includes for *Outlook PSTs/OSTs, Outlook Express DBXs, Microsoft Exchange EDB Parser, Lotus Notes, AOL, Yahoo, Hotmail, Netscape Mail* and *MBOX archives.*

6.8. FINALeMAIL

FINALeMAIL [15] can recover the e-mail database file and locates lost e-mails that do not have data location information associated with them. FINALeMAIL has the capability of restoring lost e-mails to their original state, recover full e-mail database files even when such files are attacked by viruses or damaged by accidental formatting. It can recover E- mail messages and attachments emptied from the 'Deleted Items folder' in *Microsoft Outlook Express, Netscape Mail,* and *Eudora*.

6.9. Sawmill-GroupWise

Sawmill-GroupWise [16] is a GroupWise Post Office Agent log analyser which can process log files in *GroupWise Post Office Agent* format, and generate dynamic statistics from them, analysing and reporting events. It can parse these logs, import them into a *MySQL, Microsoft SQL Server, or Oracle database* (or its own built-in database), aggregate them, and generate dynamically filtered reports, through a web interface. It supports *Window*, *Linux*, *FreeBSD*, *OpenBSD*, *Mac OS, Solaris*, other *UNIX*, and several other platforms.

6.10. Forensics Investigation Toolkit (FIT)

Forensics Investigation Toolkit (FIT) [17] is content forensics toolkit to read and analyse the content of the Internet raw data in *Packet CAPture (PCAP)* format. *FIT* provides security administrative officers, auditors, fraud and forensics investigator as well as lawful enforcement officers the power to perform content analysis and reconstruction on pre-captured Internet raw data from wired or wireless networks. All protocols and services analysed and reconstructed are displayed in readable format to the users. The other uniqueness of the FIT is that the imported raw data files can be immediately parsed and reconstructed. It supports case management functions, detailed information including *Date-Time, Source IP, Destination IP, Source MAC,* etc., *WhoIS* and *Google Map* integration functions. Analysing and reconstruction of various Internet traffic types which includes e-mail (*POP3, SMTP, IMAP*), Webmail (Read and Sent), *IM* or *Chat* (*MSN, ICQ, Yahoo, QQ, Skype Voice Call Log, UT Chat Room, Gtalk, IRC Chat Room*), File Transfer (*FTP, P2P*), *Telnet, HTTP* (Content, Upload/Download, Video Streaming, Request) and Others (*SSL*) can be performed using this toolkit.

6.11. Paraben (Network) E-mail Examiner

Paraben (Network) E-mail Examiner [18] has comprehensive analysis features, easy bookmarking and reporting, advanced Boolean searching, searching within attachments, and full *UNICODE* language support. It supports *America On-line (AOL), Microsoft Outlook (PST, OST*), *Thunderbird, Outlook Express*, *Eudora, E-mail file (EML*), *Windows* mail databases and more than 750 *MIME* Types and related file extensions. It can recover deleted e-mails from *Outlook (PST*), *Thunderbird*, etc. ***Network E-mail Examiner*** [http://www.paraben.com/network-email-examiner.html], can thoroughly examine *Microsoft Exchange (EDB), Lotus Notes (NSF),* and *GroupWise* e-mail stores. It works with *E-mail Examiner* and all output is compatible and can easily be loaded for more complex tasks.

According to Simson L. Garfinkel [19] current forensic tools are designed to help examiners in finding specific pieces of evidence and are not assisting in investigations. Further, these tools were created for solving crimes committed against people where the evidence resides on a computer; they were not created to assist in solving typical crimes committed with computers or against computers. Current tools must be re-imagined to facilitate investigation and exploration. This is especially important when the tools are used outside of the law enforcement context for activities such as cyber-defence and intelligence. Construction of a modular forensic processing framework for digital forensics that implements the "Visibility, Filter and Report" model would be the first logical step in this direction.

7. Related Work

The term "Computer Forensics" science deals with the preservation, identification, extraction and documentation of computer evidence, and like any other forensic science, relates law and science and was coined back in 1991 [20]. Kara Nance et al [21] have proposed six categories of Digital forensics including Network Forensics. Tools and Techniques for E-mail forensics fall under the category of network forensic. Many studies have been carried out for analyzing tools and techniques used in network forensics [22, 23, 24, 25] which also include e-mail forensics tools and techniques.

8. Conclusion

E-mail is a widely used and highly distributed application involving several actors that play different roles. These actors include hardware and software components, services and protocols which provide interoperability between its users and among the components along the path of transfer. Cybercriminals forge e-mail headers or send it anonymously for illegitimate purposes which lead to several crimes and thus make e-mail forensic investigation crucial. This paper portrays e-mail actors, roles and their responsibilities. It illustrated logical e-mail architecture and underlining various core components, modules and protocols used in the system. It presents the meta-data contained in e-mail message and various techniques used for e-mail forensics. The paper also introduces several software e-mail forensic tools that have functionalities to automatically analyse e-mail and produce reports providing diverse information about it.

References

[1] Suzuki, S., Nakamura, M. (2005). "Domain Name System—Past, Present and Future", IEICE Transactions of Communication, E88b (3), pp. 857-864.

[2] 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, pp. 545–554.

[3] 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, November, 1999.

[4] Crocker, D. (2009). "Internet Mail Architecture", RFC 5598, July 2009. http://tools.ietf.org/pdf/rfc5598.pdf.

[5] Internet Assigned Numbers Authority (IANA), http://www.iana.org/assignments/port-numbers

[6] Resnick P, Ed. (2001). "Internet message format", Internet Engineering Task Force (IETF); 2001. RFC 2822.

[7] Marwan Al-Zarouni. (2004). "Tracing E-mail Headers", Proceedings of Australian Computer, Network & Information Forensics Conference on 25th November, School of

Computer and Information Science, Edith Cowan University Western Australia 2004, pp. 16-30.

[8] eMailTrackerPro, http://www.emailtrackerpro.com/

[9] EmailTracer, http://www.cyberforensics.in

[10] Adcomplain, http://www.rdrop.com/users/billmc/adcomplain.html

[11] Aid4Mail Forensic, http://www.aid4mail.com/

[12] AbusePipe, http://www.datamystic.com/abusepipe.html

[13] AccessData's FTK, http://www.accessdata.com/

[14] EnCase Forensic, http://www.guidancesoftware.com

[15] FINALeMAIL, http://finaldata2.com

[16] Sawmill-GroupWise, http://www.sawmill.net

[17] Forensics Investigation Toolkit (FIT), http://www.edecision4u.com/FIT.html

[18] Paraben (Network) E-mail Examiner, http://www.paraben.com/email-examiner.html

[19] Simson L. Garfinkel, (2010), "Digital forensics research: The next 10 years", Digital Investigation, Vol. 7, pp. 64-73, doi:10.1016/j.diin.2010.05.009.

[20] New Techno logies Inc. "Computer Forensics Defined". http://www.forensics-intl.com/def4.html.

[21] Kara Nance, Brian Hay, Matt Bishop. (2009). Digital Forensics: Defining a Research Agenda, Proceedings of the 42nd Hawaii International Conference on System Sciences – 2009.

[22] Arthur, K. K., & Venter, H. S. (2004). An Investigation into Computer Forensic Tools. ISSA. Pretoria: Information and Computer Security Architectures (ICSA) Research Group.

[23] Natarajan Meghanathan, Sumanth Reddy Allam and Loretta A. Moore. (2009). Tools and Techniques For Network Forensic, International Journal of Network Security & Its Applications (IJNSA), Vol .1, No.1,April 2009.

[24] Matthew Geiger. (2005), Evaluating Commercial Counter-Forensic Tools, 2005 Digital Forensic Research Workshop (DFRWS), New Orleans, LA.

[25] Himal Lalla, Stephen V. Flowerda. (2010). Towards a Standardised Digital Forensic Process: E-mail Forensics, Proceedings of the 2010 Information Security for South Africa (ISSA 2010) Conference 2 – 4 August 2010, Sandton Convention Centre, Sandton, South Africa, http://icsa.cs.up.ac.za/issa/2010/Proceedings/Research/05_paper.pdf

Authors

M. Tariq Banday did his M. Sc., M. Phil. and Ph. D. Degrees from the Department of Electronics, University of Kashmir, Srinagar, India in 1996, 2008 and 2011 respectively. He did advanced diploma course in computers and qualified UGC NET examination in 1997 and 1998. At present he is working as Sr. Assistant Professor in the Department of Electronics & Instrumentation Technology, University of Kashmir, Srinagar, India. He has to his credit several research publications in reputed journals and conference proceedings. He is a member of Computer Society of India, International Association of Engineers and ACM. His current research interests include Network Security, Internet Protocols and Network Architecture.