

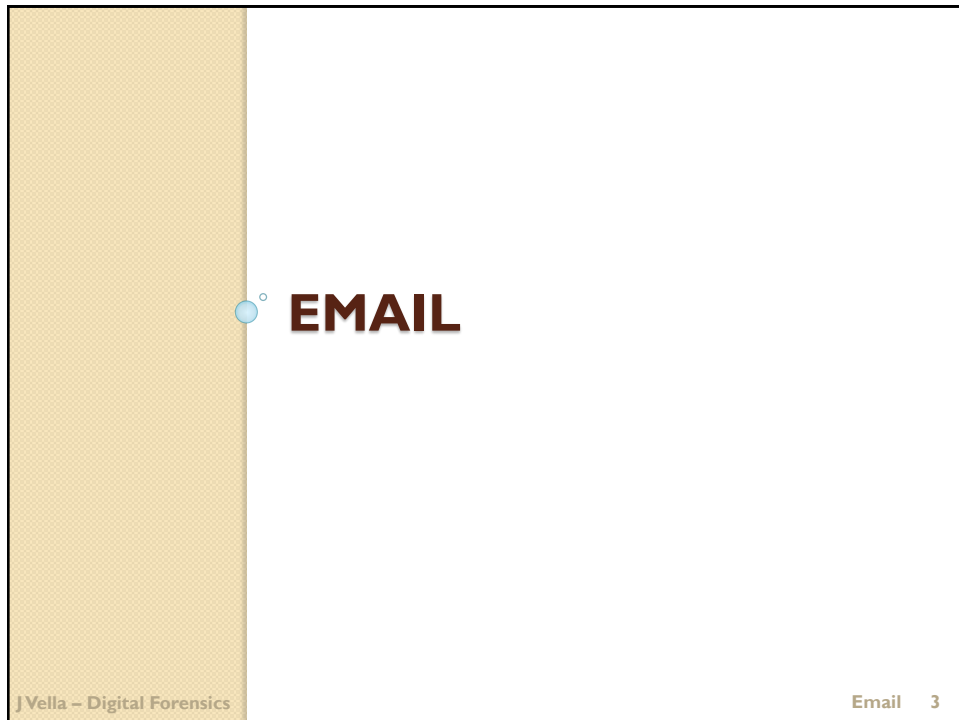
Email Forensics

°

# EMAIL FORENSICS

J Vella – Digital Forensics

Email 2



## Emails

- **Emails** are easily classified as one of the most popular communication medium.
  - Email is rated as a killer app.
  - Email entails many applications and protocols:
    - **Email client**
    - **Email server**
    - Client and server comms – popular protocol being **POP**
    - Server to server comms – popular protocol being **STMP**
    - IP and Name resolution servers:
      - **Regional Internet Registers (RIR)** for managing IP addresses;
      - **Domain Name System (DNS)** is a distributed name service for Internet \*names\* and the resolving of any name to IP location queries.
- Have a long history – as early as the sixties.
  - Ray Tomlinson is credited with introducing the at sign (@) to direct mail.
- Emails systems and protocols are regulated by **Internet Engineering Task Force (IETF)** through a sequence of stages **Requests For Comments (RFC)**, **Internet Draft** and **Internet Standard**.

J Vella – Digital Forensics

Email 4

## Email Investigations

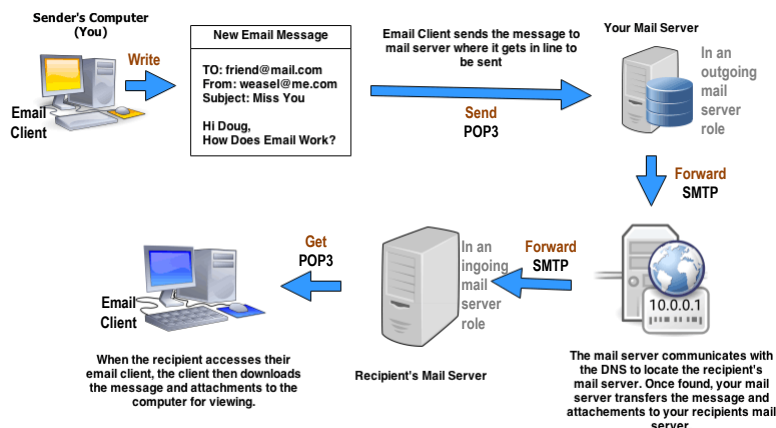


- Email communications can be abusive, a fraud, instigate a crime, solicit a crime, etc.
- Emails communication can be, sometimes too easily, forged or tempered.
  - **Email spoofing (i.e. is the creation of email messages with a forged sender address)** is now un-acceptable;
    - 💡 It was not always the case, as users currently logged onto a system which has its own SMTP and addresses can still send emails with his other email address as its send identifier – this is often called legitimate spoofing.
    - 🚫 Most installations of SMTP do not allow **open relay** mode.
- In many cases, it's not acceptable to delete Emails to hide one's action.
- **Case Studies:**
  - Oliver North, 1980s, deleted his Emails to destroy any traces of his dealing with Iran and Contras group.
    - The White House Email server still had the messages.
  - Traces of Email origin and trail are sometimes held in its header.
    - MS A Kornblum, searching for phishers John Doe (i.e. Jayson Harris), MSN attack 2005
  - Successful attacks on Email servers can enable third parties to manage and edit passing Email messages from and to its users.
  - Anti-forensics: A sender client can pass an Email through a anonymiser that can effectively cut an Email's trail.



## EMAIL SYSTEMS

## How it works!



J Vella – Digital Forensics

Email 7

## Overview of an Email message

- An array of RFCs define and regulate Email messages and messaging services.
  - Starting from RFC5322, and a number of Multipurpose Internet Mail Extensions (MIME) related RFCs.
    - Historically the RFC 733 (Apranet) and RFC 822 are the oldest.
- An Email message has two parts
  - Message header is the first;
  - Message body is the second.
- Basically the header is composed of a series of fields, and each field is identified by a printable character string and is delimited from its data payload with a colon (:) and takes a physical line.
 

```
FROM: jv@myown.email.org
```
- The body is composed of 8 bit ASCII (but clean 7 bit compliance still required) message. MIME allows for the body to include other composition methods – e.g. RTF is used in Microsoft Email clients.

J Vella – Digital Forensics

Email 8

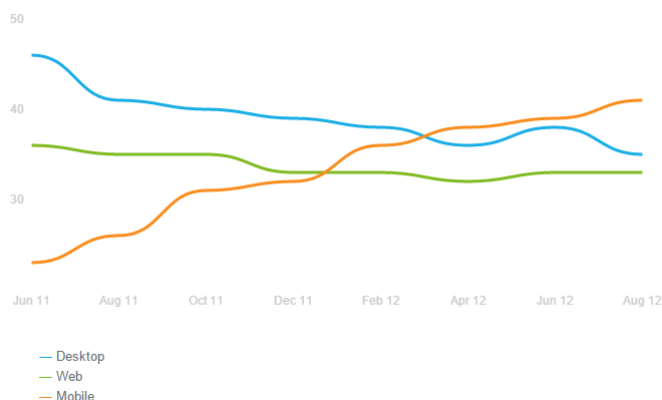
## Types of Email Clients and Mail Boxes

- Email client must write, read and manage emails for the end user. Furthermore it transparently needs to communicate with its mail server to send and get messages.
- There are two generic architectures for email clients:
  - Local, e.g. Outlook
  - Web-based, e.g. Gmail
- A Email user's messages can be stored:
  - Locally,
  - On Email server associated with user,
  - And both.
  - Furthermore traces of an email message and even email message itself can be stored, even if temporarily, on other Email servers.
- There are a number of repository formats for Email boxes:
  - Mbox;
  - Maildir;
  - Proprietary – e.g. Microsoft.

J Vella – Digital Forensics

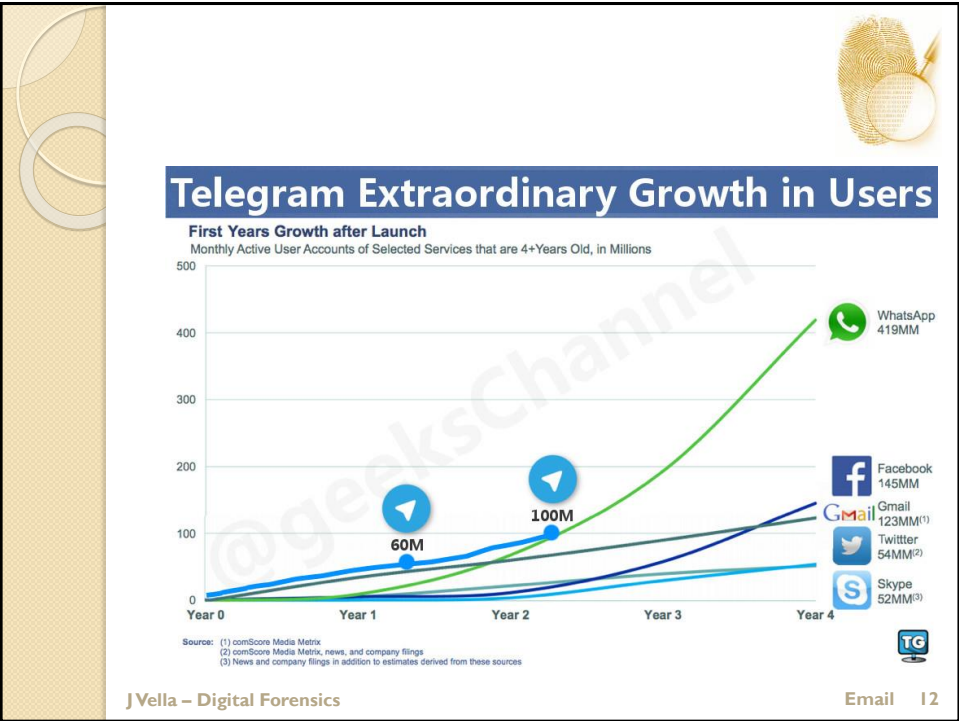
Email 9

<https://www.campaignmonitor.com/dev-resources/will-it-work/email-clients/>



J Vella – Digital Forensics

Email 10



# Client to Email Server Protocols

- Email client must interact with it Email servers:
  - Outgoing
  - Incoming
- A number of negotiation protocols exist:
  - POP (post office protocol);
  - IMAP (Internet message access protocol);
  - MAPI (MS mail API);
  - HTTP.

Protocol Service	Protocol	Details / Issues
Messages leave Server on GET mail	POP	Investigation for messages content at client's local store. (Note: POP does allow for Email messages to stay on server).
Messages stay on Server	IMAP, MAPI	Investigation for messages content at client local store, server, or both.
Web-based send and get	HTTP	Got and Send messages at server; local remnants possible to find. Some users also opt to store their messages locally. Other issues with this system – e.g. Spoof identity issues.

J Vella – Digital Forensics

Email 13

# Email Server to Server Protocols

- A popular protocol is Simple Mail Transfer Protocol (SMTP):
  - This is a straightforward protocol.
  - And each step and exchange usually leaves a (limited) trace in the message header section.
- But SMTP has:
  - Very little authentication checks, and even less is (can be) recorded.
  - Very little is done to explain and verify delivery status of an Email.
    - In fact not even delivery reports are enabled by some SMTP as spamming effectively make an user mbox filled with these!?(If not a way for spammers to identify \*live\* Email identifiers).
- It's very difficult to replace SMTP even if better proposals have been proposed and are agreeable.
- In the mean time SMTP has been reinforced with a number of techniques to address its weaknesses. E.g.
  - Sender Policy Framework,
  - Grey listing of suspicious emails.

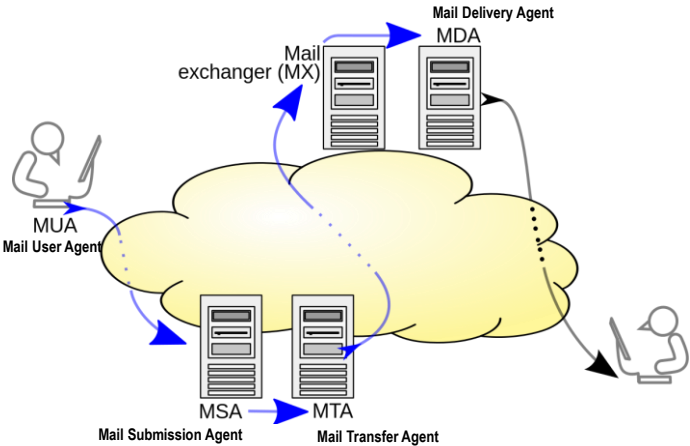
J Vella – Digital Forensics

Email 14



# SIMPLE MAIL TRANSFER PROTOCOL

## SMTP Protocol





## SMTP and Messages

- Protocol:
  - A SMTP transaction has three parts each defined with its own command:
    1. MAIL – to establish a return-path and bounce address;
    2. RCPT – to establish recipient of message. One is issued for each recipient.
    3. DATA – to earmark start of message and indicate its end. The SMTP uses two rounds for this command to pass through.
  - Each transactional command is given a compute status by the server which indicated the success or otherwise of each part.
    - 2xx – positive
    - 4xx – transient negative
    - 5xx – permanent negative
  - SMTP uses TCP port 25 as its “well known” choice.

J Vella – Digital Forensics

Email 17

## SMTP Transport Protocol Example

```

S: 220 smtp.example.com ESMTP Postfix
C: HELO relay.example.org
S: 250 Hello relay.example.org, I am glad to meet you
C: MAIL FROM:<bob@example.org>
S: 250 Ok
C: RCPT TO:<alice@example.com>
S: 250 Ok
C: RCPT TO:<theboss@example.com>
S: 250 Ok
C: DATA
S: 354 End data with <CR><LF>.<CR><LF>
C: From: "Bob Example" <bob@example.org>
C: To: "Alice Example" <alice@example.com>
C: Cc: theboss@example.com
C: Date: Tue, 15 January 2008 16:02:43 -0500
C: Subject: Test message
C:
C: Hello Alice.
C: This is a test message with 5 header fields and 4 lines in the message body.
C: Your friend,
C: Bob
C: .
S: 250 Ok: queued as 12345
C: QUIT
S: 221 Bye
{The server closes the connection}

```

J Vella – Digital Forensics

Email 18

## SMTP Header Fields

- A header field consists of a *field name* (printable ASCII characters) delineated by a colon (":"), followed by the *field body* of data, and terminated by CRLF.
- Header fields Include:
  - Originator
 

```
FROM: <mailbox+> CRLF
SENDER: mailbox CRLF
[REPLY-TO: <email+> CRLF]
```
  - Destination
 

```
TO: <email+> CRLF
[CC: <email+> CRLF]
[BCC: <email+>| [CFWS] CRLF]
```

    - Note: CFWS – comment folding white space
  - Identification
 

```
[MESSAGE-ID: identifier CRLF]
[IN-REPLY-TO: identifier CRLF]
[REFERENCES: identifier CRLF]
```

    - Although optional this is highly recommended
    - Highly recommended in replies and value payload include original message id and domain.

J Vella – Digital Forensics

Email 19

## SMTP Header Fields (continued)


- Header fields include:
  - Informational
 

```
SUBJECT: text CRLF
COMMENTS: text CRLF
KEYWORDS: < text + > CRLF
```
  - Resent (a separate set appended for every resend)
 


```
RESENT-DATE: datetime CRLF
RESENT-FROM: <mailbox+> CRLF
RESENT-SENDER: mailbox CRLF
RESENT-TO: <email+> CRLF
RESENT-CC: <email+> CRLF
RESENT-MESSAGE-ID: identifier CRLF
```

J Vella – Digital Forensics

Email 20




## Example Email Header




```
Received: (qmail 20564 invoked from network); 5 Jan 2006 16:11:57 -0000
From: foo<foo@foo.com>
To: bar@bar.com
Subject: Test
User-Agent: KMail/1.9
MIME-Version: 1.0
Content-Disposition: inline
Date: Thu, 5 Jan 2006 16:41:30 +0100
Content-Type: text/plain; charset = "iso-8859-1"
X-Originating-IP: [216.119.20.3]
Message-Id: <200601051641.31830.foo@foo.com>
X-HE-Spam-Score: 0.0
X-HE-Virus-Scanned: yes
Status: OR
Content-Length: 124
Lines: 26
```

J Vella – Digital Forensics

Email 21



## Another Example Email Header



```
Delivered-To: paul.friedman@gmail.com
Received: by 10.12.174.216 with SMTP id n34csp2326299qvd;
  Wed, 1 Feb 2017 00:39:09 -0800 (PST)
X-Received: by 10.28.27.14 with SMTP id b14mr1702258wmb.82.1485938349292;
  Wed, 01 Feb 2017 00:39:09 -0800 (PST)
Return-Path: <reply@activetrail.com>
Received: from i2.a01.ms18.atmailsvr.net (i2.a01.ms18.atmailsvr.net.
[91.199.29.18])
  by mx.google.com with ESMTPS id
  5si23398790wrr.176.2017.02.01.00.39.08
  for <paul.friedman@gmail.com>
  (version=TLS1_2 cipher=ECDHE-RSA-AES128-GCM-SHA256 bits=128/128);
  Wed, 01 Feb 2017 00:39:09 -0800 (PST)
Received-SPF: pass (google.com: domain of reply@activetrail.com designates
91.199.29.18 as permitted sender) client-ip=91.199.29.18;
Authentication-Results: mx.google.com;
  dkim=pass header.i=@activetrail.com;
  spf=pass (google.com: domain of reply@activetrail.com designates
91.199.29.18 as permitted sender) smtp.mailfrom=reply@activetrail.com;
  dmarc=fail (p=NONE sp=NONE dis=NONE) header.from=gingersoftware.com
X-IADB-IP: 91.199.29.18
X-IADB-IP-REVERSE: 18.29.199.91
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/relaxed; q=dns/txt;
d=activetrail.com; s=at; h=X-BBounce:X-IADB-URL:Sender:Submitter:X-
Feedback-ID:From:To:Date:Subject:MIME-Version:Content-type:Content-
Transfer-Encoding; bh=GytDyTyADleCfGk0d7bL4F2bXbTuWeb/xtpIVvVaCRw=;
b=agh6nUFjt5FC7rBC2BwXFullNuG+k14R7bBsstb4erjtzfTn4z/NPHNhVb4AxlYXoOgX+
Il6n58CcTckwQdmxpxt/BzFjWVziBdzU1WichHhPabVFeKctyp6pCjv4+d2FVIIeuxqi
v5dBtOjJXBVpOwU0mqgRceh3pgcqv5Rj4=
```

J Vella – Digital Forensics

Email 22

## Email Header with Tracing Indications

- Trace (added for every message passing through SMTP)
  - Ensure that the FROM field datum has a name (a/c) and address (e.g. IP address) as per HELO conversation between Email client and Email Server conversation;
  - The FOR field may contain a list of path entries when multiple recipients are indicated in a message.
- An Email server must not change a RECEIVED field previously added.
- A Email server must add, just prior to present RECEIVED, its own RECEIVED field.
- If Email server is the final mail server then it must insert the return-path line in the beginning of the mail data section.

J Vella – Digital Forensics

Email 23

## Tracing SMTP Emails

- The key point of tracing SMTP email is that we can trust all data put into the message after it left the control of the sender (but excluding the IP address, since that could be spoofed), but nothing before.
- To identify user one uses details from the header FROM field and check it out with WHOIS server.
  - Clearly IP addresses can also point at sources and bounce points.
  - E.g. Email addresses make sense – check with Email Dossier
  - E.g. Domain names make sense - check with Domain Dossier.  
<https://centralops.net/co/>

J Vella – Digital Forensics

Email 24

## Tracing SMTP Emails (continued)

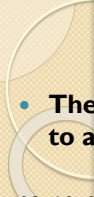


- How to identify a spoofed message?
  - Mostly content!
  - Header fields, if contradictory, point at mishandling.
    - E.g. different IPs in the received and X-Originating-IP line;
    - E.g. known mail servers, e.g. GMAIL, IP number do not seem right (ie after comparing to known legitimate messages);
  - Continuation in REVEIVED lines seems suspect.
  - SMTP servers have logs too;
    - These can be compared to emails purporting to be emanating from, or passing through, them.
  - Time stamps issues (even if one caters for different timezones!?).
- Tools, even online exists, that automate initial investigations – eg.
  - <http://www.cyberforensics.in/OnlineEmailTracer>
  - <http://www.iptrackeronline.com/email-header-analysis.php>

## Email Server Logs



- Log data:
  - Email content;
  - Sending email message IP address;
  - Receiving and reading data timestamps;
- Deleted files, representing email messages, can be recovered from Email server temporary filing space.
- Management and config of Sendmail (on Unix boxes):
  - `/etc/sendmail.cf` -- config details and settings
  - `/etc/syslog.conf` -- which sendmail events to log
  - `/var/log/maillog` -- SMTP & POP3 comms details with IPs and dates



## Explanation of SMTP log file


- The transactions in the log when an email is received and processed to a local user with no errors:

```
03:19 03:22 SMTPD (00180250) [192.168.1.131] connect 209.221.59.70 port 2539
03:19 03:22 SMTPD (00180250) [209.221.59.70] EHLO jetbn.net
03:19 03:22 SMTPD (00180250) [209.221.59.70] MAIL FROM:<info-jjgcdshx@infostreet.us>

03:19 03:22 SMTPD (00180250) [209.221.59.70] RCPT To:<user@domain.com>
03:19 03:22 SMTPD (00180250) [209.221.59.70] C:\IMail\spool\D28de0018025017cd.SMD
3827


03:19 03:22 SMTP- (00000260) processing C:\IMail\spool\Q28de0018025017cd.SMD
03:19 03:22 SMTP- (00000260) ldeliver mail.domain.com user (1) <info-jjgcdshx@infostreet.us> 2354

03:19 03:22 SMTP- (00000260) finished C:\IMail\spool\Q28de0018025017cd.SMD
status=1
```



J Vella – Digital Forensics

Email 27



## Explanation of SMTP log file

- At this point your mail server establishes a connection and receives the email:

Date	Time	Process	Connection #	Server IP	Connecting Server	Port #
03:19	03:22	SMTPD	(00180250)	[192.168.1.131]	connect 209.221.59.70	port 2539

Connecting Server Name

```
03:19 03:22 SMTPD (00180250) [209.221.59.70] EHLO jetbn.net
```


Who is Sending the email

```
03:19 03:22 SMTPD (00180250) [209.221.59.70] MAIL FROM:<info-jjgcdshx@infostreet.us>
```

Who the email is for


```
03:19 03:22 SMTPD (00180250) [209.221.59.70] RCPT To:<user@domain.com>
```

File created for this message in the spool	Msg Size Bytes
03:19 03:22 SMTPD (00180250) [209.221.59.70] C:\IMail\spool\D28de0018025017cd.SMD	3827



J Vella – Digital Forensics

Email 28



## Explanation of SMTP log file

- From this point forward your server has the message and is processing it for delivery to the mailbox of the recipient:

What Process is doing File that is being processed


```
03:19 03:22 SMTP- (00000260) processing C:\IMail\spool\Q28de0018025017cd.SMD
```

Who received the email Who sent the email Deliv. Size


```
03:19 03:22 SMTP- (00000260) 1 deliver mail.domain.com user (1) <info-jjgcdshx@infostreet.us> 2354
```

SMTP finishes processing the email Status of Process

```
03:19 03:22 SMTP- (00000260) finished C:\IMail\spool\Q28de0018025017cd.SMD status=1
```



J Vella – Digital ForensicsEmail 29



## Explanation of SMTP log file

- The transactions in the log when a message is processed and sent to a remote user with no errors:

Date Tme Server IP Process Conn# Server Connecting toServerIP Port

```
20030325 102649 127.0.0.1 SMTP (2292) Connect todomain.com [65.45.210.46: 25] (1)
```

Recipient Server Hello Message

```
20030325 102649 127.0.0.1 SMTP (2292) 220 What? Mail for Me??? X1
```

Sending user name


```
20030325 102649 127.0.0.1 SMTP (2292) >EHLO fromdomain.com
```

Recipient Server Hello Response

```
20030325 102650 127.0.0.1 SMTP (2292) 250-mail.todomain.com says hello
SMTP (2292) 250-SIZE 0 // SMTP (2292) 250-8BITMIME // SMTP (2292) 250-DSN //
SMTP (2292) 250-ETRN // SMTP (2292) 250-AUTH LOGIN CRAM-MD5 // SMTP (2292) 250-AUTH=LOGIN // SMTP (2292) 250 EXPN
```

Who sent the message

```
20030325 102650 127.0.0.1 SMTP (2292) >MAIL FROM:sender@fromdomain.com
20030325 102650 127.0.0.1 SMTP (2292) 250 ok
```



J Vella – Digital ForensicsEmail 30

## Explanation of SMTP log file



### Who is to receive the message

```
20030325 102650 127.0.0.1 SMTP (2292) >RCPT To:<recipient@todomain.com>
```

### Receiving server's permission to accept mail

```
20030325 102650 127.0.0.1 SMTP (2292) 250 ok its for recipient@todomain.com
```

### Beginning of message data send

```
20030325 102650 127.0.0.1 SMTP (2292) >DATA
```

```
20030325 102651 127.0.0.1 SMTP (2292) 354 ok, send it; end with <CRLF>.<CRLF>
```

### End of data send

```
20030325 102651 127.0.0.1 SMTP (2292) >.
```

### Message Queued for delivery

```
20030325 102651 127.0.0.1 SMTP (2292) 250 Message queued
```

### Message received by server and placed in recipients mailbox

```
20030325 102651 127.0.0.1 SMTP (2292) rdeliver todomain.com recipient@todomain.com  
(1) <sender@fromdomain.com> 356
```

### Closing connection to receiving server

```
20030325 102651 127.0.0.1 SMTP (2292) >QUIT
```

J Vella – Digital Forensics

Email 31

## SYSLOG.CONF example



```
$ grep mail /etc/syslog.conf
```

```
# Log anything (except mail) of level info or higher.
```

```
*.info;mail.none;authpriv.none;cron.none  
/var/log/messages
```

```
# Log all the mail messages in one place.
```

```
mail.* /var/log/maillog
```

- Explanation:

- No Email server messages/alerts are to be collected!
- Send all mail **messages** to /var/log/maillog
- Default log level is 9 – which is info level of syslog. For example I5 will log all incoming SMTP commands.

J Vella – Digital Forensics

Email 32



Investigating email related cases



## FORENSIC TOOLS

## Desiderata



- What if you have to choose an tool set/kit for email forensics?
  - What features would you like to have as a start?
- Able to read a diverse list of email file formats:
 

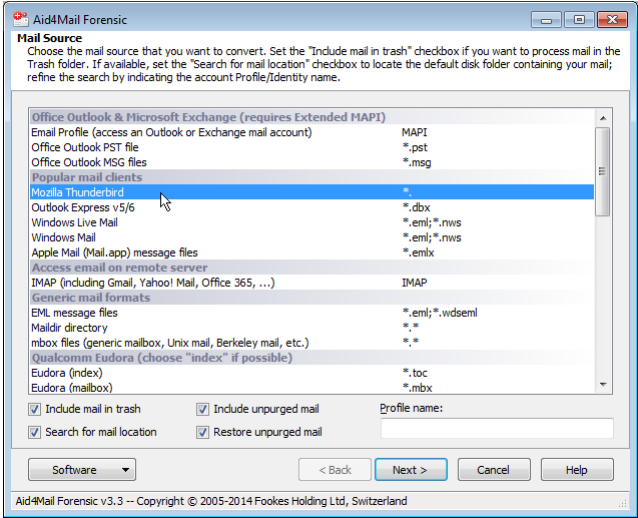
(Even if an organisation adopts one format you want to take advantage of tool set knowledge).  
Also you can avoid having to rely on email client software that is not meant for forensic investigations.
- Search and retrieval capabilities:
  - Expressing queries
    - individual, collective, and building a sequence
    - expressions based on Boolean predicates, regular expression based predicates, association rules, sequence rules
    - Multi language?
- Speed (and related to the previous) – indexing the mail box?
- Management of results:
  - tagging and bookmarking of hits
  - Logging of all interaction (especially if queries are being run on site)
- Deleted email messages search (i.e. search email server log files too).
- Case management (including preparation of Court required statements).
- Training programme for the tool set and an operational base deployed 24x7.

# Specialised Email Forensic Tools

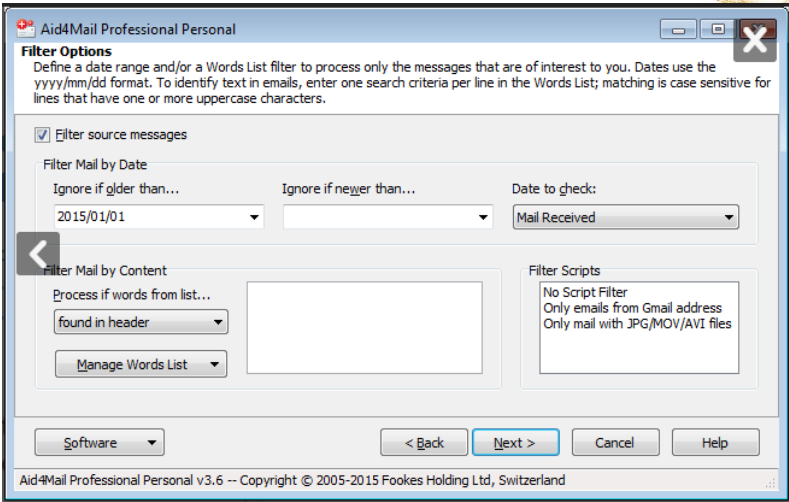


- Open Source: Add4Mail, Paraben Email Examiner etc
- Add4Mail:
  - Read Mboxes:
    - Local storage, and off-site through IMAP.
  - Search Mboxes:
    - By field, and content through regular expressions.
  - Email statistics and content indicated.
  - Ported for Windows, Mac OSX and Linux.
  - Exports – pretty and mbox conversion too.

# Add4Mail - Screenshots



# Add4Mail - Screenshots

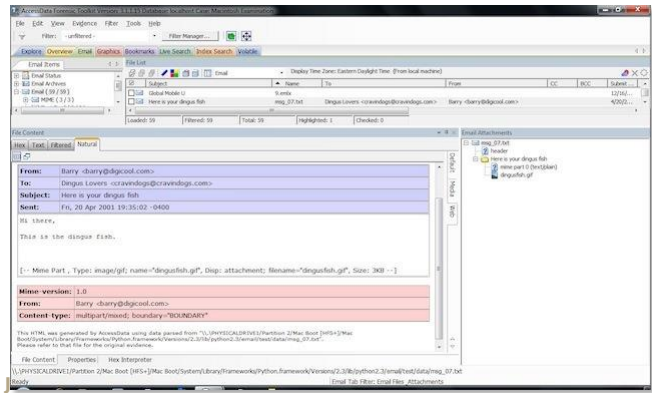


J Vella – Digital Forensics

Email 37

# Specialised Email Forensic Tools

- Commercial products to aid digital forensics investigation for Emails.
  - AccessData FTK to recover email messages
  - One component is the email viewer.
    - Can read various mbox formats.
    - Email attachments are also viewable.
    - Filtering options available – even as hex strings.



Email 38

## Extracting Attachments from Mboxes

- Recall our attempts to extract embedded JPEGs from a Word DOCX file – ie in file carving.
  - Use very similar techniques to extract files attached in Emails.
  - Remember, some, Mboxes, are non-ASCII and proprietary – but a Hex Editor works wonders.

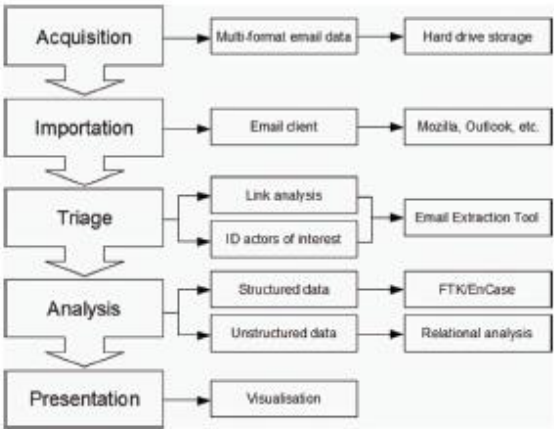
## EMAIL AT LARGE

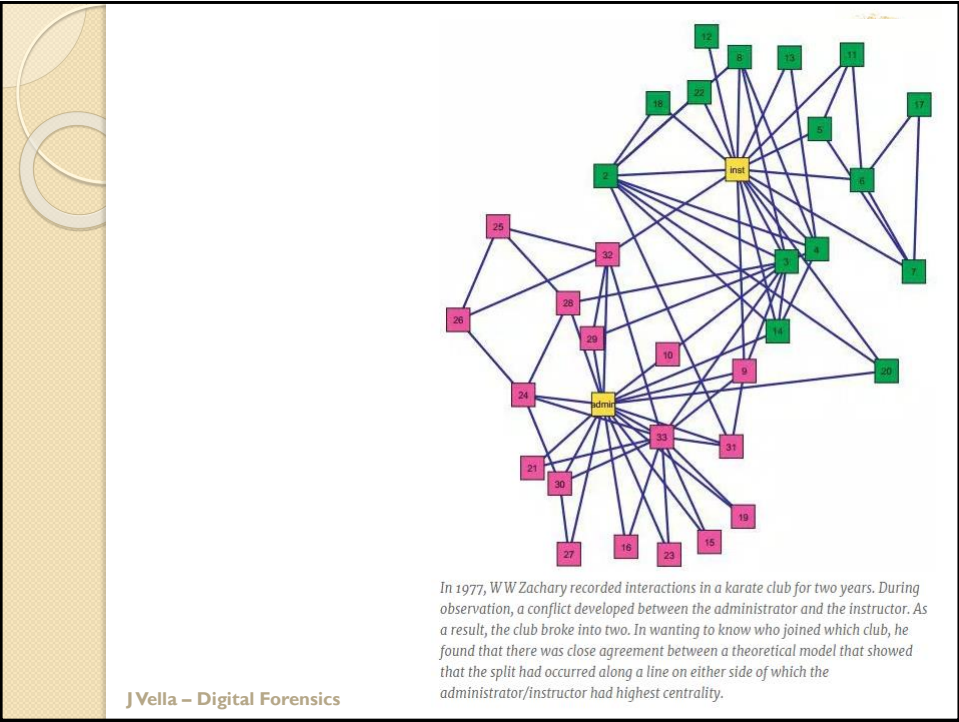
# Studying a body of Email Messages



- We have looked, up to now, at an email message level.
- What about a person’s mailbox?  
What about a company’s mailbox?
- How can we extract and elucidate the unstructured data, such as relationships within the email network, power relations or network bridges that may be a key concern to a forensics investigation?
  - Email investigation over a batch of messages can yield significant trends and contacts that are associated with an event of interest.
  - Moreover the email trails and exchanges can identify the events and their dependencies that lead to the event of interest.
- Nonetheless this is not an easy endeavour:
  - E.g. data and servers are spread widely.
- Common techniques on Email boxes include:
  - Clustering of emails by a “distance” measure;
  - Process model from email exchanges.

# A possible framework





(Not so easy nowadays but still employed in more sophisticated attacks )



# EMAIL SPOOFING

## Spoofing – What, Why and How



- Spoofing is the process that forges, or changes, an Email's message sender address (or identifier).
  - Spam and Phishing email attacks use spoofing methods to hide the attack intention from the recipients.
- The cause of spoofing has much to do with authentication between SMTP servers.
  - And checking if SMTP have authority to execute a user's request.
- The basic technique is to alter the header's addresses data fields.
  - The first forwarding and exchange would have:
    - **Mail Form** – Usually the Email address presented to a recipient on *Return-Path*.
      - Although checks on validity of address are done, none are made on the server's mandate to send email on behalf of sender.
    - **Recipient** – The Email address of message final destination.
  - The second exchange, if Oked by SMTP server, the rest of message is sent together with From and Reply-to fields.
    - No effective authentication is done.
  - Clearly the recipient reads details, even the From and Reply-to fields, which are not 100% secure.

J Vella – Digital Forensics

Email 45

## Primitive Spoofing through SMTP



- One attacking technique is to telnet an SMTP using:

```
$ telnet hermes.not.uom.ac.mt 25
220 hermes ESMTP Sendmail vXX Date & Time
helo xx.xx.xx.xx
250 ... Pleased to meet you
mail from: notme@not.uom.ac.mt
250 ... Sender ok
rcpt to:dpresident@not.uom.ac.mt
250 ... Recipient ok
data
354 Enter mail, end with a period
Date: 01 April 1999 10:15:11
From: theladyinblack@not.uom.ac.mt
To:dpresident@not.uom.ac.mt
Subject: Promotion
Is this this easy!
Have a nice day.
xxx
.
250 ... Message accepted for delivery
quit
...
```



- Look at <https://www.youtube.com/watch?v=gf6j0H9lBYw>
- It's a 12 minute video called "Sending mail using telnet and SMTP authentication" by Gopal Thrope (up to Windows 7) – last updated Nov 2011.

J Vella – Digital Forensics

Email 46