Adding digital forensic readiness to the email trace header

F.R. Van Staden and H.S. Venter
Department of Computer Science
University of Pretoria
Pretoria, South Africa

Abstract— **The protection strategies proposed and implemented to protect users against spam, focus on specific areas that need to be protected e.g. Anti-Spam filters that protect the user's mailbox from bulk unsolicited email. Digital forensics is based on scientifically proven methods to collect and analyze digital information. Employing digital forensic techniques to gather and analyze email information provides a new dimension to the fight against spam.**

Adding digital forensic readiness to email will allow for the gathering of forensic information. The digital forensic information can be used to verify information contained in the trace header of an email. The authors propose augmentations to the receive header, that is part of the trace header, currently specified for SMTP to implement digital forensic readiness.

Incorporating digital forensics, adds a level of integrity to the trace header information that can be used for other purposes e.g. creating a spam detection mechanism or tracing the origin of spam. Digital forensic information is added to the email envelope so there is no effect to the content of the email. Therefore, the content remains untouched.

The authors examine the addition of digital forensic information and highlight the changes that will need to be implemented in the SMTP trace header. The authors propose the gap detection algorithm that is used to find gaps in the received-tokens of the received header. The information that is generated by the gap detection algorithm is also discussed. In conclusion, the addition of digital forensic readiness adds a level of integrity to the SMTP trace header that can be used to add a level of trust.

Keywords-SMTP; Spam; Digital forensics; Digital forensic information; Digital forensic readiness;

I. INTRODUCTION

SMTP was intended to be a lightweight protocol to standardize electronic communication over networks. Since the standardization of the Simple Mail Transfer Protocol in 1982[1], the utilization of SMTP based communication has increased. As a communication protocol SMTP is reliable and easy to use. Being such an easy protocol to understand and use, it is also prone to abuse.

The most common protection strategies proposed and implemented to protect users against spam, include amongst others, Anti-Spam filters that protect the user's mailbox from bulk unsolicited email. Digital forensics is based on scientifically proven methods to collect and analyze digital information. Employing digital forensic techniques to gather and ANALYZE email information, however, provides a new dimension to the fight against spam.

The authors propose augmentations to the current trace header to implement digital forensic readiness. Incorporating digital forensics, adds a level of integrity to the SMTP trace header information that can be used for other purposes e.g. tracing the origin of spam. Digital forensic information is added to the email envelope so there is no change to the content of the email.

The background section provides a brief overview of SMTP and digital forensics. As part of the SMTP overview, email spoofing and the trace header are briefly discussed. As part of the digital forensics background, digital forensics and digital forensic readiness are discussed. After the background section, the authors discuss the proposed augmentation to add digital forensic information to the trace header. Next the authors discuss a gap detection algorithm that is used to find gaps in the received-tokens of the received header. Finally the information generated by the gap detection algorithm is discussed.

II. BACKGROUND

SMTP and digital forensics are discussed as part of the background section.

A. SMTP

The Simple Mail Transfer Protocol, known as SMTP, was first proposed in the Request for Comments (RFC) 821[1] in August of 1982. The protocol is based on the "snail" mail architecture where electronic mail is sent from one "post office" to the next until the mail is delivered to the intended "mailbox". SMTP describes the "envelope" and not the content of the email. The basic rule of SMTP is: as soon as an SMTP host acknowledges delivery of the SMTP envelope, it is that SMTP host's responsibility to deliver the envelope to the correct email box.

RFC 1425 [2] accepted in February 1993, described a way to extend the services SMTP offers, so that calling clients can ask what services are available on the server. SMTP Service

978-1-4244-5494-5/10/$26.00 ©2010 IEEE

Extensions are added to the core specification as the service extension becomes more popular. RFC 821 was made obsolete in April 2001 by RFC 2821 [3].The new specification included some service extensions and updates that were in use at the time.

The latest RFC that describes SMTP is RFC 5321, released October 2008 [4]. SMTP standards must be backwards compatible, which means services that are not in regular use might not be described in later RFC documents but are expected to remain available [4].

The information in the SMTP headers is stored in clear text. Therefore the information can easily be edited. The possibility that the mail headers could have been edited makes the information in the headers suspect. The editing of the mail headers are done to hide information, like the origin of the email, from the receiving email box. The action of editing the mail headers to hide information is called spoofing and is discussed in the next section.

1) *Email Spoofing*

Email spoofing is the act of editing or falsifying the SMTP header information to hide the true origin or root of an email [5][6]. Spoofing is also used to add fake validity to the content of an email by using a well known and trusted domain as the originating domain in order to perpetrate a phishing attack. RFC 4406 [17] is an experimental RFC that describes two tests for SMTP servers to perform in order to verify that a mail header has not been spoofed.

The first test is the Purported Responsible Address (PRA) test [7]. Lyon [7] describes a way to try and find the PRA inside the SMTP headers. If no PRA can be found, the email has a high probability of being spoofed. If the PRA can be established it is still not proof that the SMTP header has not been spoofed, since the address used, for the PRA, is the first well formed address the PRA algorithm found. The PRA needs to be tested further to establish its validity.

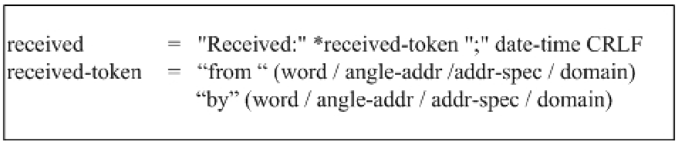
The second test uses a Sender Policy Framework (SPF)[8] to authenticate if a SMTP client is allowed to act on behalf of the originating domain. Wong [8] proposes the SPF as a method to detect a spoofed email that uses valid domain information to appear legitimate. The supposed sender domain and the routing information in the header is authenticated by the DNS of the domain owner to determine if the SMTP client's domain has the authority to act on behalf of the supposed sending domain. If the DNS returns a failed authentication, the email is marked as possibly spoofed. The next section looks at the current specification for the trace header in SMTP and what it is currently used for.

2) *SMTP trace header*

The trace header consists of two sub headers namely the return-path and received headers. The return path header is used to store the address where error reports should be sent. The received header stores the delivery path with a data stamp for each delivery entry. The format of the trace header is defined in RFC 5322 [9] as the trace rule. The trace rules are defined in Augmented Backus-Naur Form (ABNF) which is defined in STD0068 [10].

The usage of the trace header is defined in RFC 5321 [4] for delivering error reports to the sender as well as to create delivery reports that can be used as input information when doing trouble shooting. Klensin [4] proposes in RFC 5321 that the trace header should be made compulsory for all SMTP servers that implement the RFC 5321 standard.

The augmentation that will be discussed later is defined for the received header, therefore only the received rule for the received header is given. The received rule and the received-token rule are shown in figure1.



```
received          =  "Received:" *received-token ";" date-time CRLF
received-token    =  "from " (word / angle-addr /addr-spec / domain)
                     "by" (word / angle-addr / addr-spec / domain)
```

Figure 1Received rule and received-token rule

The received rule in figure1 indicates that the received header must start with the word Received followed by a possible empty list of received-tokens. The list of received-tokens is followed by a date-time stamp and a Carriage Return Line Feed (CRLF) character that indicates the end of the received header entry.

The received-token rule indicates that the received-token must start with the word "from" followed by the address of the sending host. The received-token ends with the word by followed by the receiving host's address. The next section defines digital forensics and looks at digital forensic tracing.

B. *Digital Forensics*

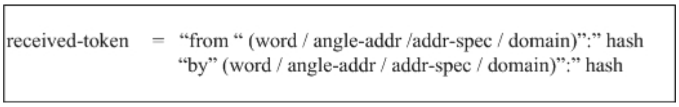
Digital forensic science is a relitivly new field of study that evolved from forensic science. According to the Oxford Dictionary [11], digital forensic science is the systematic gathering of information about electronic devices that can be used in a court of law. Digital forensic science is more popularly called digital forensics and sometimes also called computer forensics. Palmer [12] defines digital forensics as " the use of scientifically derived proven methods towards the presirvation, collection, validation, identification, analysis, interpretation, documentation, and presentation of digital evidence derived from digital sources for the purpose of facilitation or furthering the reconstruction of events". Palmer's definition describes the digital forensic process whereas Oxford describes digital forensic science. The Digital Forensic Process Moddle (DFPM) by Kohn, et al. [13] states that "any digital forensic process must have an outcome that is acceptable by law".

Rowlingson [14] defines digital forensic readiness as consisting of two objectives. The first objective is to maximise the environment's capability of collecting digital forensic information. The second objective is to minimize the cost of a forensic investigation.

Preparing SMTP to be digital forensically ready, a mechanism will need to be added, to SMTP, to preserve, collect or validate the information contained in the SMTP headers. The information gathered from the SMTP headers can then be used as part of a digital forensic investigation The next section discusses the addition of digital forensic information to the received header.

III. Augmenting the Trace Header with Digital Forensic Information

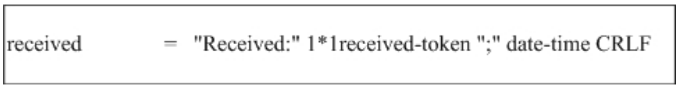
The authors propose that digital forensic information is added to the SMTP trace header to verify the validity of the trace information according to digital forensic techniques. The digital forensic information augmentation is focused on the received rule that is a sub header in the SMTP trace header. The received rule is changed to store the digital forensic information by adding a hash value to the received-token. The augmented receive-token rule is shown in figure2.



```
received-token    =  "from " (word / angle-addr /addr-spec / domain)":" hash
                     "by" (word / angle-addr / addr-spec / domain)":" hash
```

Figure 2 Augmented Receive-token Rule

The augmented receive-token rule depicted in figure2 shows the addition of two hash values. The first after the "from" line that contains the value of the sending domain hashed together with the DNS lookup IP of the sending domain. The second after the "by" line that contains the value of the host's domain hashed together with the DNS lookup IP address of the host's domain. The hash value can be created using SHA-1 or a similar hashing function to preserve the integrity of the received-token. The augmented receive rule is shown in Figure3.



```
received          =  "Received:" 1*1received-token ";" date-time CRLF
```

Figure 3 Augmented Received Rule

The received rule depicted in Figure 3 is changed to contain at least one and only one received-token entry. The received header must contain at least one received token so that the digital forensic information is always present. The received header contains only one received-token entry per line to simplify the information extraction process that is used for the gap detection algorithm. The next section discusses the gap detection algorithm.

IV. Gap Detection Algorithm

The receive header contains a list of received-tokens that are in the form of sender host and receiver host pairs also called the send-receive-pair. The list might not be complete if all the SMTP hosts, which are used during the sending process, do not update the received header as per the specification. This will mean that gaps in the list could exist. The gaps can be detected by using the send-receive-pairs.

The gap detection algorithm is used to detect gaps in the digital forensic information stored in the received header and to store the filtered tracing information for later use. The send-receive-pair shows one send-receive sequence during the sending process of the mail. The authors define a gap as a break in the receiving header when one or more send-receive pairs are not detected in the digital forensic information. The gap detection algorithm is defined as follows:

- Step 1. Store the received send-receive pairs, from last entry to first, in a queue.
- Step 2. Remember the receiving host address of the first entry in the queue.
- Step 3. Look at the next send-receive pair and compare the sending hosts address with the stored receiving host address. If the addresses are the same, proceed to step 4. If the addresses are not the same store the sending address and proceed to step 5.
- Step 4. Store the receiving host address in the proven list and proceed to Step 6.
- Step 5. Store the receiving host address in the proven list. Store the sending host address in the gap list and set the gap found flag to true. Proceed to step 6.
- Step 6. If the next send-receive pair is null proceed to Step 7, else store the next receiving host address and proceed to Step 3.
- Step 7. If the gap found flag was set to true end the algorithm with the output message "gap detected" else end the algorithm with the output message "no gap found".

At the end of the algorithm two lists exist: the proven list showing the hosts that implement the augmented receiving header and the gap list showing the last known hosts that did not implement the augmented received header. The next section discusses what the proven list and the gap list can be used for.

V. Utilizing the Information in the Stored Lists

The gap detection algorithm produces two lists, as discussed before. The information stored in the lists can be used in one of two ways. If there were no gaps detected for a specific email, the information in the list pertaining to the email can be used to digital forensically trace the origin of the email by following the verified send-receive-pairs in the received header.

Although the digital forensic information is added to the received header, the header is still stored in clear text and can still be edited by an adversary. It is therefore accepted that only when there are no gaps in the received header can the send-receive-pairs be used to trace the origin of spoofed email.

The second use of the stored list only works with large data sets that were gathered from email that originate from many different email domains. The information is still valuable even if gaps were detected in the received header. Using the information in the proven list an SMTP network is generated showing the SMTP forwarding lines that are known to implement the Received header with the digital forensic information. The SMTP network can be called the SMTP trust network, meaning that the information in the receiving header of emails, which travel through the SMTP trust network, is more trustworthy. By analyzing the created SMTP trust network and by adding the information in the gap list, gaps that continually occur can be identified. Depending on the size of the data set, paths that bypass the gap areas in the network can be created. SMTP servers that form part of the SMTP trust network can be given a trust value indicating that the SMTP server is a preferred path for mail forwarding. Email forwarded by trusted SMTP servers can be given a trust value as well. The trust value assigned to email can be used to determine how much anti-spam resources must be assigned to the email to detect if the email is spam.

The creation and assignment of the trust values are out of the scope of this article. Future work will include defining the process of assigning a trust value to the SMTP server and the email.

VI. Conclusion

The article proposes the addition of digital forensic information to the email trace header. The proposed addition focuses on adding the digital forensic information to the received header. The hash value is used to verify the domain information in the received-token that is contained in the received header.

The proposed gap detection algorithm uses the received-token information to create two information lists. The proven list stores information about SMTP hosts that implements the proposed trace header. The gap list stores information about SMTP hosts that appear not to implement the proposed trace header.

The two lists can be used to create a SMTP trust network. The SMTP trust network can in turn be used to assign a trust value to an email. The trust value is used to determine how much anti-spam resources must be applied to the email to detect if the email is spam.

Future work will focus on the refinement and implementation of the SMTP trust network and its uses.

VII. References

[1] Postel, J B. Simple Mail Transfer Protocol. RFC 821. s.l. : Internet Engineering Task Force, August 1982.

[2] Klensin, J, et al. SMTP Service Extensions. RFC 1425. s.l. : Internet Engineering Task Force, February 1993.

[3] Klensin, J. Simple Mail Transfer Protocol. RFC 2821. s.l. : Internet Engineering Task Force, April 2001.

[4] Klensin, J. Simple Mail Transfer Protocol. RFC 5321. s.l. : Internet Engineering Task Force, October 2008.

[5] Network-Dictionary. http://www.networkdictionary.com/security/. http://www.networkdictionary.com. [Online] http://www.networkdictionary.com, 2009. [Cited: 27 April 2009.] http://www.networkdictionary.com/security/b.php.

[6] Allman, E and Katz, H. SMTP Service Extension for Indicating the Responsible. [Electronic]. s.l. : Internet Engineering Task Force, 2006.

[7] Lyon, J. Purported Responsible Address in E-Mail Messages. RFC 4407. s.l. : Internet Engineering Task Force (2006), April 2006.

[8] Wong, M and Schlitt, W. Sender Policy Framework (SPF) for Authorizing Use of Domains in E-Mail, Version 1. RFC 4408. s.l. : Internet Engineering Task Force (2006), April 2006.

[9] Resnick, P. Internet Message Format. RFC5322. s.l. : Internet Engineering Task Force (2008) October 2008.

[10] Crocker, D. Ed. and Overell, P. Augmented BNF for Syntax Specifications: ABNF. STD0068-RFC5234. January 2008.

[11] University Press, Oxford. Possible entries for. http://www.oup.com. [Online] [Cited: 22 April 2009.] http://www.oup.com/oald-bin/web_getald7index1a.pl.

[12] Palmer, G.L. Road Map for Digital Forensic Research. [Electronic Publication] s.l. : Digital Forensic Research Workshop (DFRWS), Digital Forensic research workshop, 2002.

[13] Kohn, M. Eloff, J.H.P. and Olivier, M.S. UML Modelling of Digital Forensic Process Models (DFPMs). [Document] Pretoria : Information and Computer Security Architectures (ICSA) Research Group University of Pretoria, 2009.

[14] Rowlingson, R. A Ten Step Process for Forensic Readiness. International Journal of Digital Evidence. 2004. Vol. II, 3.

[15] Oxford Dictionary. AskOxford.com. AskOxford.com. 2010.

[16] Rousseau, D.M., et al. Not so different after all: A cross-discipline view of trust. Academy of management review. 1998, Vol. 23, 3, pp. 393-404.

[17] Lyon, J and Wong, M. Sender ID: Authenticating E-Mail. RFC 4406. s.l. : Internet Engineering Task Force (2006), April 2006.

[18] FIPS. fip180-1.htm. http://www.itl.nist.gov/fipspubs/. [Online] Federal Information Processing Standards Publication, 17 April 1995. [Cited: 29 04 2010.] http://www.itl.nist.gov/fipspubs/fip180-1.htm. 180-1.

[19] InternetNews. Report Says Spam Arms Race Escalating. http://www.ironport.com/. [Online] IronPort In the News, 16 March 2009. [Cited: 16 March 2009.] http://www.ironport.com/company/pp_internet_news_03-16-2009.html.

[20] Ramsdell, B. and Turner, S. Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 3.2 Message Specification. RFC 5751. January 2010.