Global Information Assurance Certification Paper

Sender Authentication To Reduce E-mail Spoofing

Simon C. Tremblay
GIAC Security Essentials Certification
Practical Assignment
Version 1.4b Option 1
August 2, 2004

1. Abstract

One of the weakest links of the e-mail system is the lack of a general sender authentication system. Not having a proper identification of senders makes it easy to forge e-mail which in turn is helping many e-mail born attacks.

To better understand this, we will briefly describe the workings of SMTP, the main protocol used for e-mail interchange. Then we will define spoofing and some of the mail based attacks that are facilitated by the lack of positive ID. Finally, we will outline what is being done by the Internet community, in particular, two new standard that were presented as Internet Draft and that are being reviewed by the Internet Engineering Task Force: Sender ID from the MARID group and DomainKeys presented by Yahoo!.

2. Introduction

Internet E-mail as we know it, like the Internet itself, has evolved from a communication tool for tech savvy researchers to a mundane mean of communication used by high school students and Fortune 500 executives alike. The uses have evolved, but the underlying technology is basically the same, and the means of abuse abound. Why is that? What attacks are enabled by mail forgery? What is being done to correct this and can it really help?

We will try to answer these questions in the next few pages.

3. What is SMTP

SMTP, Simple Mail Transfer Protocol is a high level protocol that allows mail messages to be carried between systems. It was first defined in 1982 in RFC821. (Postel)

3.1. Overview of the protocol

SMTP is widely used on the Internet and is considered “the de facto standard for email transmission across the Internet”.(wikipedia) Several Mail Transfer Agents (MTA) implement the specs of RFC821 and RFC2821 defining this application level protocol. It was designed to run above various transmission subsystems (Postel pars. 1) but is so widely used with TCP that the revised RFC2821 focuses only on this transport protocol and refers to the original RFC821 for other transport model (Klesin).

SMTP is a point to point protocol using TCP port 25. It allows forwarding within a domain and relaying to different domains. For example, forwarding would allow a company to setup a limited number of Internet exposed mail gateways that can then forward mail to appropriate servers on their domain. Relaying would permit an ISP to handle mail from different domains with a single server.

When sending to a fully qualified domain name address, which is most often the case, SMTP will rely on DNS mail exchange (MX) records to resolve the receiving server for a given domain. (Klensin pars.5)

SMTP is text based and can be easily tested by manually sending SMTP commands with a Telnet program. Because it is text based, binary files have to be encoded to be moved with it.

3.2. Typical steps when sending messages.

For example, here is the typical route taken by an e-mail sent by an Internet home user:

1. The user uses is favorite mail client to prepare his message. The mail includes a recipient address and a sender address.
2. The client software establishes a connection to his ISP's outgoing mail server on port 25 and transfers the message.
3. The ISP server looks up the DNS MX record of the recipient's domain. From there, the ISP server becomes the client to the machine specified by the MX information.
4. If the destination server accepts the connection, the message is sent along. The server can refuse sender is allowed, based on IP and/or mail domain of sender or an invalid recipient address.
5. If recipient server is the final destination, it will hold the message until it is retrieved, typically through POP3 protocol (port 110).

This is the most common and basic Internet mail route. It may differ slightly, mostly at the client ends. For example, Lotus Notes and Microsoft Exchange natively do not use SMTP and POP3 to communicate with client software. They will though use an SMTP agent when sending mail across the Internet.

4. E-mail Spoofing and E-mail attacks

In computer science, the general definition of *spoofing* is sending forged information so that it appears to come from a third party, usually with the intent of abusing the trust relationship between the two original parties. Besides masquerading as a trustworthy party for malicious intent, spoofing makes it difficult to trace the originator and that in turn relieves the sender of all accountability.

IP packets can be spoofed using simple programs like Hping2 (Cole 8-15). This technique is often used in various hacker attacks. Forging a complete SMTP message with spoofed IP is harder because a hacker would have to forge the session without receiving the answers from the server. and the complete IP sequence numbering would have to be guessed.

Also web sites can be spoofed, hosting really believable look-alike pages, usually to obtained sensitive information from users. Web site spoofing is not related to the SMTP weakness we are looking into, but is more a web browser issue or an end user education issue. Still, combined with forged mails it becomes a devastating phishing attack, as explained later.

What we are more concerned about is e-mail spoofing, the sending of forged e-mail as it facilitates many attacks. The SMTP protocol's lack of authentication mechanism makes this relatively easy. In a standard transaction, the client machine should identify

itself in the HELO command. This information is included in the received path, along with the IP address used for the connection. Also, the sender's email address has to be specified in the "MAIL FROM:" command. This is specified in the RFC, but basic SMTP does not provide any means of validation for this information.

Spoofed headers will give anonymity to the sender, which is most often used to hide from retaliation and even sometime prosecution when conducting illicit activities through e-mail.

When John Klensin edited the revised SMTP specification RFC2821, he recognized that spoofing was a problem and concluded with the following remark:

> "This specification does not further address the authentication issues associated with SMTP other than to advocate that useful functionality not be disabled in the hope of providing some small margin of protection against an ignorant user who is trying to fake mail." (Klensin pars. 7.1)

Already in 2001, e-mail spoofing was an issue, but the problems associated with it had not yet reached the scale it has today. Now, people trying to fake mail are not ignorant users but crafty criminals. Taking action against these problems is now at the forefront mainly because the community of users is losing confidence in the Internet as a whole and e-commerce in particular.

Here are explained the most prominent attacks enabled by e-mail spoofing, namely SPAM, Phishing, Brand Name Spoofing and Worm Propagation.

4.1. SPAM

Historically the first SPAM message is reported to be "the Green Card Spam" (Everett-Church) and was sent in 1994 to Usenet newsgroups. What started as a nuisance and a lack of netiquette has now grown into a serious security problem, in particular due to the ever growing volume of SPAM. (Symantec Corporation) On its web site, Symantec Corporation states that in June 2004, 65% of mail traffic handled by their Brightmail product was reported as SPAM.

SPAM represents a security risk because it affects availability of mail infrastructures, global and privates. It uses up expensive bandwidth, server processing power and storage space, both as disk space and back-up handling. It also carries high indirect costs in lost productivity for corporations. Trend Micro, using a conservative 40% of mail as SPAM, evaluates an average cost of 188$ per employee, assuming that they waste 3 seconds per piece of SPAM mail they receive. (Trend Micro SPAM Calculator)

In his GSEC practical paper, Adalberto Zamudio discusses at length the most accurate definition of SPAM. The difficulty in establishing a definitive definition arises because of the suggestive nature of SPAM content. Some message considered unwanted by one user might be quite valuable to someone else.

Focus of this document is not to find that definitive answer, but here are some basic criteria based on personal experience by which most average Internet user would recognize SPAM:

- *Mail was not specifically requested or agreed upon*; special offers from online vendors may not be SPAM if the user specifically subscribed to this periodical.
- *Mail is sent repeatedly and cannot be stopped*; the continuous flow of messages is what annoys most users, especially when it cannot be stopped due to lack of opt-out mechanism and/or invalid sender information, making it impossible to write back a complaint.
- *Mail content cannot be guessed without opening the message itself*; to deceive users further, senders name and subject lines are often intentionally meaningless and users have no choice but to open the message to evaluate its relevance.
- *Return path for dead mail is invalid*; this overloads mail infrastructure as mail servers try repeatedly to send back undeliverable mail to non-existent return addresses.

4.2. Malware Propagation

On July 29, 2004, A quick survey of Trend Micro Top Ten Threats (see Appendix A) showed that more than half of them propagated by e-mail. Once a machine is infected, the malware harvests e-mail addresses on the machine and then mass mail itself in a forged e-mail.

The messages are randomly generated to avoid filtering. Often, the sender's address and the message content are socially engineered to trick the user in opening the mail and the attached documents. They will generally be crafted along one of the two most popular themes:

- From: <Bob> Bob@hotmail.com. Bob (Everyone knows a Bob) is sending you something cool to check out… really friendly, it sparks your curiosity and compulses you to open the attachment. (see WORM_NETSKY.C)
  or
- From: Support@YourISP.net, the support from your ISP is asking you to perform a task to help them out. Just follow the instructions in the attached zipped document… represents authority, very polite and calls to your kindness and the need to do good. (see WORM_MYDOOM.M)

For faster propagation, many of these will have their own SMTP engine and will be able to connect directly to remote servers to send themselves. Obviously when the From: is spoofed, the infected machine is not a legitimate server to send message for the YourISP.net domain, for example.

4.3. Phishing and "Brand Spoofing"

Phishing is a type of e-mail fraud that is currently growing fast. In fact, according to a June 2004 Gartner research quoted in ZDNet "phishing, has grown into the fastest growing form of consumer theft in the United States". (Hines)

A typical Phishing attack will combine socially engineered spoofed e-mail and spoofed web site. The e-mail is the lure, hence the fishing analogy, to attract the victims to the web site where private or financial information will be asked from them. Faking communication from a major financial institution or e-commerce site, mail and web page will look quite like the real site they pretend to be from, with the same layout, color scheme, logo etc.

Depending on the information collected from the victim, phishers will then proceed to empty bank accounts, max out credit cards, obtain new credit card or even get a loan to buy a car. Basically, with enough information, they could do anything from bank fraud to complete identity theft.

Phishing by spoofing well established company brand name also has a negative effect on the reputation and the trust consumers have in the company being spoofed. Considering the marketing money spent by large corporations on their image, this has become a major concern to them and they should swiftly adopt any proposal that can help protect those investments. It is much harder to regain lost trust than it is to maintain.

5. Current SPAM Fighting Techniques

SPAM is being fought on many fronts, including the legal front. The legal battle is quite convoluted due to the global reach of the Internet and the difficulty to establish clearly jurisdiction boundaries. For this reason, we will focus on the most common technical solutions.

Since SPAM is viewed as a problem, many products have been designed to reduce the volume of unwanted mail, including phishing attacks and e-mail born malwares.

Blocking or tagging these mails can be done at the mail server or at the client's end, but no matter where mail is sorted, it will use a variation of two main techniques, often both: black lists and content filtering.

5.1. Blacklists

Local blacklist is one of the earliest and simplest way to identify spammers. Maintain a list of known spamming e-mail addresses, servers IPs, domains or host names and refuse connection when they try to establish SMTP transaction. It is hard to keep up to date so DNS Blacklists were developed.

DNS Blacklist works like the local blacklist in such that it will deny connection from hosts specified by IP or by name in a special list. The difference is that the list you now use is usually maintained by a community of mail administrators and updated almost in real time. Then these lists are made available through a DNS database.

Although simple, blacklists now have a limited effect on SPAM volume. The difficulty comes from the fact that spammers now constantly use different servers to circumvent that type of protection. Also many recent worms have established a wide network of

infected machines onto which they've installed mail relay software. Quite often these are "always on" PCs with a broadband connection that can relay volume of mail similar to that of corporate mail servers. With these army of zombies, it does not matter if you block one of them, has many other infected machines are available to carry out the next "delivery".

Because of the lack of authentication in SMTP, there was no way of deciding that such and such IPs were legitimate aol.com servers. Even if it's efficiency is limited, blacklists can be used to supplement other more sophisticated and CPU-intensive techniques, like content filtering.

5.2. Content filtering

Content filtering is scanning the content of each mail message for signs of unwanted material. Here again the techniques have evolved as users and spammers continue the cat and mouse game.

These techniques range from simply looking for banned keywords in subject lines to statistical analysis, through weighted regular expressions analysis of formatting and punctuation. Keyword filtering is now mostly ineffective because it is not flexible enough to catch all variants of a banned word.

Other advanced techniques like regex and statistical baysian filters are more efficient, but they require more processing power and are then more "expensive" and as the volume of spam grows constantly, this as become more and more of an issue.

Another challenge in content filtering is teaching a computer how to differentiate legitimate mail from unwanted ones. It is easy for a human to recognize unwanted mail but much harder to define it in a way for a machine to detect.

6. Improving SMTP with Sender Authentication

Like mentioned previously, action has to be taken in order to restore confidence in the Internet and e-commerce. To that end, majors ISPs have come together and formed the Anti-Spam Technical Alliance, or ASTA:

> "The Anti-Spam Technical Alliance (ASTA) is a collaborative effort between six leading Mailbox Providers and the Internet community to establish technical and non-technical solutions for handling unwanted and unsolicited e-mail (spam)." (ASTA)

The Alliance, founded by America Online, British Telecom, Comcast, EarthLink, Microsoft, and Yahoo!, released in June 2004 its "Statement Of Intent" to jointly fight SPAM in general and spoofing in particular. The document includes an extensive set of best practices that Alliance members will implement. Many of these best practices should also be implemented by other ISPs and any corporations with an Internet presence.

In the “Curbing E-mail Forgery” section of the document, two types of technologies are presented to help identify and authenticate senders and are called “IP address” and “content signing” approaches. These map out to specific protocols already presented by Alliance’s members. Microsoft/pobox.com Sender ID is an “IP address” approach while Yahoo’s DomainKeys is a “content signing” solution. Both of these have been presented to the IETF as Internet draft proposals.

Here is a brief explanation of those two propositions and the benefits they can bring to the community.

6.1. Sender ID

Sender ID is a merge of two earlier technical proposition, pobox.com Sender Policy Framework (SPF) and Microsoft Caller ID for e-mail. The purpose of the specification is:

> “Given an email message, and given an IP address from which it has been (or will be) received, is the SMTP client at that host address authorized to send that email message?” (MARID pars. 3)

Following the Sender ID specification, when a MTA receives a message from a host, it retrieves the “e-mail policy” for the domain that appears in the “from:” header of the mail to basically determine if the sending host is authorized to send mail for the said domain.

The specification explains how to publish in a DNS TXT record the domain “e-mail policy” document. That “e-mail policy” document is an XML infoset that at first will include a description of which hosts are allowed to send e-mail on behalf of the domain. The XML schema of the policy can be expanded to allow for future features; any information pertaining to e-mail could be included in it, as suggested by the designer. For backward compatibility, the specification also allows support for a Sender Policy Framework TXT DNS record. Simpler in form, the basic features of the SPF record are similar to the e-mail policy document, but it does not offer the same extensibility as the XML e-mail document.

Using DNS to publish the e-mail policy document or the SPF record leverages the pervasive DNS infrastructure of the Internet. MTAs already rely heavily on DNS for determining the destination hosts of each e-mail; adding this DNS operation to a mail transaction will not add much delay as the e-mail policy of the most frequent destinations will be cached locally, just like MX records and host IPs.

The specification also describes what status to return based on the information contained in the e-mail policy document and the sender IP and suggests the typical actions that should be taken. The possible results are "pass", "fail", "softFail", "neutral", "transientError", "hardError" or "none".

Mail administrators should review these status codes and should determine their own policy based on these results, keeping in mind that they do not give any indication about the legitimate *content* of the mail but just qualifies the authorization of the sending host

to send mail for this domain. Until the specification gains wide acceptance, a basic policy would be to reject all messages that return a “fail” status, refuse “transientError” but return an SMTP 450 transient error code and analyze all other mail with other anti-spam tools, like black lists validation and content filtering.

Sender ID has the merit of being simple to implement and at minimal costs, using already existent infrastructure. This should favor a rapid deployment, which is desired to take full advantage of it. E-commerce sites and financial institutions, which are the more vulnerable to spoofing attacks, should be early adopters and thus the public will be protected from the more costly scams.

It should become a de facto standard in MTA implementation so that mail administrators can change their policy to reject everything that does not specifically return a “pass” result. Until then, the measure will not be fully effective because spammers can avoid being blocked by it by avoiding spoofing domains that publish Sender ID records.

Nevertheless, pre-filtering messages with Sender ID should reduce the load on the down path validation tools by weeding out obvious forgeries and it will increase the accuracy of black lists by forcing the domain included in the from: header to match with an authorized domain.

6.2. DomainKeys

DomainKeys is the Content Signing proposal presented by Yahoo! Like Sender ID, it uses the DNS infrastructure to publish domain validation information.

> "DomainKeys" creates a domain-level authentication framework for email by using public-key technology and the DNS to prove the provenance and contents of an email.” (Delany Abstract)

Under the DomainKeys framework, an MTA will sign a message using the private key of a public/private key pair and insert the signature in the message as a header. After receiving a signed message, a receiving host will retrieve the public key from the pair to validate that the message really comes from the domain it pretends to be from. The result of this validation will also be included in the message headers.

In the initial draft of DomainKeys, the public/private key pair are RSA keys, but the specification includes provisions for using different signing algorithms. The same is true for key distribution which is initially proposed through DNS but with provisions for alternate distribution mechanism. An “e-mail policy” similar to Sender ID is also published through DNS. The “e-mail policy” associated with DomainKeys does not specifies which hosts can send for a domain, but will explain what to do with messages that fail verification. The policy outlined in the draft is presented as an interim measure until a more robust e-mail policy specification is introduced, specifically by the MARID group of the IETF. The MARID group is the one who presented Sender ID internet Draft, which with its XML structure will easily accommodate all the requirements of DomainKeys.

Using DNS is recognized as an advantage because if the domain owner has control over his DNS records, he also has full control over public key management for his domain at no costs.

The Internet draft includes a description of how to publish the public-key representation in DNS. It also includes an explanation of how the signature header should be built using tag=value pairs. This allows for new tags to be introduced in future development of the specification. It then describes the status codes that should be passed down to the receiving party: "good", "bad", "no key", "revoked", "no signature", "bad format" and "non-participant".

Just like with Sender ID function results, mail administrators should review these status codes and should determine their own local policy based on these results, keeping in mind that: “DomainKeys is only intended as a "sufficient" method of proving authenticity” (Delany pars. 7.1.1).

Just like with Sender ID, DomainKeys deployment is simple and uses the already existent DNS infrastructure, which makes it a low cost solution that can be quickly deployment. Again it should become a de facto standard in MTA implementation so that local policies can be strengthen to take full advantage of DomainKeys.

And again, sites that rely on trust from the user community like financial institutions and e-commerce sites should implement this rapidly to prevent losing their customers trust.

7. Conclusion

The lack of authentication allowing e-mail spoofing is one of the weaknesses of the SMTP protocol. Sender identification can help reduce e-mail born threats. To be efficient though, a standard must be agreed upon and then widely and quickly deployed on a majority of mail servers. Partial deployment will not allow reaping the full benefits of the new protocol.

And although this will help reduce e-mail spoofing, it will not eradicate it completely. Sender authentication as to be integrated as part of a defense in depth strategy, like the one put forward by the Anti-Spam Technical Alliance (ASTA) which promotes technical solutions along with a set of best practises to tackle the problem on multiple front.

Sender ID and DomainKeys complement each other and are not SPAM prevention tools. They are enabling technologies and will help increase the effectiveness of more conventional techniques.

Both Sender ID and DomainKeys will surely help restore confidence in e-mail in particular and the Internet in general.

Appendix A

Trend Micro Top Ten Threats, July 29, 2004.

Threat Name	Type of Threat	Propagation Methods
WORM_SASSER.B	Worm	Network
PE_ZAFI.B	File Infector	E-mail, P2P
WORM_NETSKY.P	Worm	E-mail, P2P
WORM_MYDOOM.M	Worm	E-mail
HTML_NETSKY.P	Html	E-mail
WORM_RBOT.ZG	Worm	Network
WORM_NETSKY.D	Worm	E-mail
WORM_NETSKY.B	Worm	E-mail, P2P
WORM_NETSKY.Z	Worm	E-mail, P2P
JAVA_BYTEVER.A	Java	Web

Source Trend Micro. “Top Threats”. July 29, 2004.
<http://www.trendmicro.com/vinfo/default.asp?sect=TT>

References

In Print

Cole, Eric. SANS Security Essentials Cookbook Version 2.2.SANS Institute, 2004.

Internet

ASTA. “Anti-Spam Technical Alliance Technology and Policy Proposal”. June 22, 2004. July 29, 2004. <http://docs.yahoo.com/docs/pr/pdf/asta_soi.pdf>

“Brightmail - Spam Percentages and Spam Categories”. Symantec Corporation. July 11, 2004. July 29, 2004. <http://www.brightmail.com/spamstats.html>

Delany, Mark. “Domain-based Email Authentication Using Public-Keys Advertised in the DNS (DomainKeys)”. Internet Draft. May 18, 2004, July 29, 2004. <http://www.ietf.org/Internet-drafts/draft-delany-domainkeys-base-00.txt>

“DomainKeys: Proving and Protecting Email Sender Identity”. Yahoo!. July 29, 2004 <http://antispam.yahoo.com/domainkeys>

Everett-Church, Ray. "The Spam That Started It All". Wired News. Apr. 13, 1999. July 29, 2004. <http://www.wired.com/news/politics/0,1283,19098,00.html>

Hines, Matt. "Gartner: Phishing on the rise in U.S.". *ZDNet - News Security*. June 15, 2004. July 1, 2004. <http://zdnet.com.com/2100-1105_2-5234155.html>

Klensin, John C. "RFC 2821 - Simple Mail Transfer Protocol". Apr. 2001. July 1, 2004. <http://www.faqs.org/rfcs/rfc2821.html>

Leblanc, Charlene."Slippery Slope or Terra Firma Current and Future Anti-Spam Measures". SANS InfoSec Reading Room. June 30th, 2003. June 26, 2004. <http://www.sans.org/rr/papers/index.php?id=1153>

MARID working group. “MTA Authentication Records in DNS”. Internet Draft. June 23, 2004. June 27, 2004. <http://download.microsoft.com/download/d/a/2/da2821f5-6acb-4058-8974-5a3c7d187794/senderid.pdf >

Postel, Jonathan B. "RFC 821 - Simple Mail Transfer Protocol". Aug. 1982. July 1, 2004. <http://www.faqs.org/rfcs/rfc821.html>

Olsen, Stephanie. “Alliance turns up heat on SPAM”. *CNet News*. June 22, 2004. July 30, 2004. <http://marketwatch-cnet.com.com/Alliance+turns+up+heat+on+spam/2100-1032_3-5243727.html?type=pt&part=marketwatch-cnet&tag=feed&subj=news>

Reardon, Marguerite. "Spam seen as security risk". *CNET News*. Feb. 11, 2004. June 21, 2004. <http://news.com.com/2100-7355_3-5157275.html>

“Simple Mail Transfert Protocol”. *Wikipedia*. Wikimedia Foundation Inc. July 29, 2004. http://en.wikipedia.org/wiki/SMTP>

Sorkin, David E. SPAM Laws. Dec. 16, 2003. July 30, 2004. <http://www.spamlaws.com/index.html>

“SPAM Calculator”. Trend Micro. July 29, 2004. <http://www.trendmicro.com/en/products/gateway/spam/evaluate/spam-calculator.htm>

“Top Threats”. Trend Micro. July 29, 2004. <http://www.trendmicro.com/vinfo/default.asp?sect=TT>

“WORM_NETSKY.C - Technical details”. *Virus Encyclopedia.* Trend Micro. Feb. 25, 2004. July 29, 2004. <http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=WORM_NETSKY.C&VSect=T>

“WORM_MYDOOM.M - Technical details”. *Virus Encyclopedia.* Trend Micro. July 26, 2004. July 29, 2004 .<http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=WORM_MYDOOM.M&VSect=T>