Mail Non-Delivery Notice Attacks

Stefan Frei[1], Ivo Silvestri[2], Gunter Ollmann[3]

[1] stefan.frei@techzoom.net [www.techzoom.net]
[2] silvestri@isitech.com [www.isitech.com]
[3] gunter@ngssoftware.com [www.ngssoftware.com]

www.techzoom.net/mailbomb

Abstract. Analysis of e-mail non-delivery receipt handling by live Internet-bound e-mail servers has revealed a common implementation fault that could form the basis of a new range of DoS attacks. Our research in the field of e-mail delivery revealed that mail servers may respond to mail delivery failure with as many non-delivery reports as there are undeliverable Cc: and Bcc: addresses contained in the original e-mail. Non-delivery notification e-mails generated by these systems often include a full copy of the original e-mail sent in addition to any original file attachments. This behavior allows malicious users to leverage these mail server implementations as force multipliers and flood any target e-mail system or account.

1 Introduction

Towards the end of 2002, the authors discovered that from time to time there were massive amounts of mail traffic destined for non existent e-mail accounts on our mail servers. This tremendous increase in traffic came without warning and lasted for one to several days, only to stop as suddenly as it started. Close examination of this traffic revealed that it consisted almost entirely of non-delivery notification (NDN) messages from any number of legitimate mail servers, mostly major Internet access providers and mail portals. The authors concluded that spammers had chosen to fake our mail domains within the reply-to addresses of the malicious spam they were sending. A large proportion of the mail accounts originally targeted by the spammers did not exist and therefore their authoritative mail servers generated NDN messages which were promptly sent to the also non existing accounts on our mail systems.

It is important to note that these spam e-mails were not directly targeted at accounts on our servers. Instead, the ***reply-to*** address of the offending spam-mails contained our some of our registered domain names. Therefore, only *NDN messages* were sent to our systems and not the original spam mail. A closer inspection of these NDN messages revealed interesting differences as to how mail-servers generate their responses. It was this analysis which instigated our research into the field of NDN attacks.

This paper analyses the methods utilized by common mail servers and gateways to generate of NDN messages and the implications for potential abuse. Through two related testing methods, experimental data is presented which was gathered from probing initially 8000+ random mail servers and then a representative sample of the Fortune 500 mail systems. This data confirms the high likelihood for future abuse and targeted denial of service (DoS) attacks against SMTP services.

2 Transport mechanism for Internet mail

2.1 SMTP mail delivery process

Internet e-mail is delivered through mail servers using the SMTP (Simple Mail Transport Protocol) service – defined in 1982 and 2001 by RFC-821 [1] and RFC-2821 [2]. The STMP protocol defines the commands that may be used by mail servers to communicate to each other in order to exchange e-mail messages. In the following examples we will discuss the SMTP communications necessary in exchanging an e-mail message from the sending mail server (*alfa*) to a receiving mail server (*bravo*). This example places special emphasis on the handling of delivery failures when sending a single message to multiple recipients on the *same* receiving host.

2.2 Mail to multiple recipients

Any modern e-mail program can send a mail message to one or multiple recipients within a single SMTP session by sending a list of recipients in the *To:*, *Cc:* and *Bcc:* address fields. When the same message is sent to multiple recipients, the SMTP protocol encourages the transmission of a single copy of the data for all recipients at the same destination (or intermediate relay) host [2].
Within a SMTP session (host to host connection), the sending server first identifies the originator of the message through the ***mail from*** *<address>* command. If the receiving server accepts the sender it then sends a *rcpt to <recipient>* response for each recipient address destined for this host. The receiving server individually accepts or rejects each recipient address by responding with *250 OK* or a *550 no such user* reply. If at least one recipient is accepted by the receiving server, the sending server issues the ***data*** command followed by the content of the message. The receiving server responds with a *250 OK* command if the content of the message was received successfully. If at this stage the receiving server rejected one or several recipients, the sending server must generate NDN messages for the rejected recipients and send these to the originator of the message.
It is this process of generating NDNs which lies at the heart of this paper and discussed in detail in the next section.

Table 1. Sample SMTP session of server *alfa* attempting to send a mail to multiple recipients at server *bravo*. Example 1 contains all valid recipients while in Example 2 some recipients are invalid.

Example 1	Example 2
> MAIL FROM:<bob@alpha.lan>	> MAIL FROM:<bob@alpha.lan>
< 250 OK	< 250 OK
> RCPT TO:<alice@bravo.lan>	> RCPT TO:<no.john@bravo.lan>
< 250 OK	< **550 no such user**
> RCPT TO:<adda@bravo.lan>	> RCPT TO:<aida@bravo.lan>
< 250 OK	< 250 OK
> RCPT TO:<aida@bravo.lan>	> RCPT TO:<no.larry@bravo.lan>
< 250 OK	< **550 no such user**
> DATA	> DATA
< 354 data;end with <CRLF>.<CRLF	< 354 data;end with <CRLF>.<CRLF
> Blah blah blah...	> Blah blah blah...
> ...	> ...
> <CRLF>.<CRLF>	> <CRLF>.<CRLF>
< 250 OK	< 250 OK
Users alice@bravo.lan, adda@bravo.lan and aida@bravo.lan exist and host bravo.lan accepts the mail for delivery. The data of the message is only sent once from host alfa.lan to bravo.lan to save bandwidth.	Host bravo.lan accepts the mail as at least one user (aida@bravo.lan) was accepted. The other users no.john@bravo.lan and no.larry@bravo.lan were rejected.

Depending on what recipients were accepted or rejected during the SMTP session, we can examine the following cases:

2.2.1 Successful Mail Delivery

If all recipients are accepted by *bravo* then the job for *alfa* is complete. *Bravo* is now required to deliver the message to the final destination, whether this be a local mailbox or forwarding on to another mail server. Should *bravo* later discover that delivery of the message to some of the recipients is not possible, the mail service must then compose and send a NDN message to the originator of the message.

2.2.2 Partly Successful Mail Delivery

Partly successful mail delivery means that during a SMTP session at least one of the recipient addresses were accepted, while the others where rejected (invalid users). For the recipients that *bravo* accepted, it takes full responsibility for subsequent delivery. For recipients the *bravo* server rejected, it is the sender server (*alfa)* that must generate NDN messages to the originator of the message since the *bravo* server refused the specific recipient by responding with SMTP *550 no such user* replies.

2.2.3 Failed mail delivery - all recipients refused

The sending server *alfa* must generate NDN messages. The *bravo* server has completed its required tasks.

2.3 Mail gateways

It is common for organizations of a certain size to employ more than one mail server for security, mail filtering, load balancing and routing reasons. Usually there is a gateway mail server designed to accept all incoming mail from the Internet. This gateway server will typically forward any inbound e-mail to the respective internal mail server or to an anti-virus and/or anti-spam filter. It is important to note that these mail gateways often accept all incoming mail for the domains of the organization, irrespective of the user, e.g. these gateways do not inherently know which addresses correspond to valid or invalid user accounts.

As per the RFC definitions, the mail server that last accepted the delivery of a message is responsible for either delivery to the final destination (relay to next host or deliver to a local mailbox) or must generate a NDN message to inform the *mail from* <*address*> originator of the delivery failure.

Unfortunately, the SMTP RFC’s only define the communication between mail servers. After accepting a message through SMTP, mail servers frequently have to cache the message in a spool-file or queue for later processing. This intermediary processing is not defined by any SMTP transmission protocols and the details depend upon the specific mail server software and configuration.

3. Non-delivery notification messages

If a SMTP server has accepted the task of relaying a message and later finds that the recipient is incorrect, or that the mail cannot be delivered for whatever reason, then it must construct a NDN message and send it to the originator of the undeliverable mail. However, the response is dependant upon the configuration and software version of the SMTP server, and consequently there are several approaches to generate these NDN messages:

3.1. Generation of NDN messages

While RFC-821 [1] requires the generation of a NDN message if an e-mail cannot be delivered to the final destination, it leaves the detail on how to compose the response open to the programmer or subject to configuration of the mail server. In the case of a single message being sent to a single recipient, the process of generating a NDN message is straightforward. However, if the message was sent to multiple recipients and delivery failed for more than one, there are response choices with subsequent delivery implications:

3.1.1 Originator of the original mail

The NDN message must be sent back to the originator of the undeliverable mail, which is determined from the SMTP session handshake - e.g. the *mail from <address>* command. This address however can easily be faked which means that someone not related to the original message could receive the NDN message from this server.

3.1.2 Content of the NDN message

The goal of the NDN message is to inform the originator that his e-mail did not reach the destination. The content of the NDN message is not defined by the SMTP RFC's. It is down to the software developer to decide upon the content of the NDN message. The following options are available when constructing the response:

1. Send just enough information to identify the mail and failed recipient and the reason for the failure.
2. Send information as above and include the original e-mail text or part of it for reference purposes.
3. Send information as above and include the complete original e-mail text and include all attachments that were sent.

3.1.3 Number of NDN messages

If the original message cannot be delivered to more than one recipient RFC-821 provides two options as to how to generate NDN message(s):

1. A single notification which lists all of the recipients that failed to get the message.
2. Separate notification messages for each failed recipient.

3.2. Issues with the generation of NDN messages

We identified three minor issues due to the lack of definition on how NDN messages should be generated. The issues could be combined and potentially lead to mass mail attacks against any nominated e-mail account by abusing key mail server failures.

3.2.1 Spoofed e-mail originator

The recipient of the NDN message can be 'spoofed' (electronically impersonated). If a malicious attacker sends a mail to *john@bravo.lan* with the faked reply-to address *alice@delta.lan,* then *alice@delta.lan* will receive a NDN message from *bravo.lan* for a message he/she never sent.

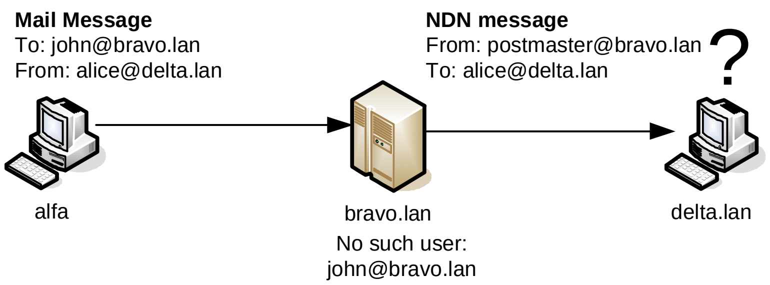

Fig. 1. Spoofed 'mail from' address flow.

3.2.2 Overly complex content of the NDN message

If the content of the NDN message generated contains a complete copy of the original e-mail message including its attachments, additional attacks against the spoofed sender may be included within the attachment.

For example, *bravo.lan* could be abused by a third-party and used to send malicious content to *alice@delta.lan* such as viruses, trojans or compression-bombs [4].

3.2.3 Multiple invalid recipients

An e-mail to several invalid recipients (included within the *To: Cc:, Bcc*: fields) could be sent to a mail server. If this mail system responds with individual NDN message for each invalid recipient, the mail server could then be misused as a force multiplier and mass mail a target e-mail address.

For example, a malicious attacker can send one e-mail to multiple *N* invalid recipients at *@bravo.lan* having the originating e-mail address faked as *alice@delta.lan*. Consequently, *alice@delta.lan* receives *N* NDM messages from *bravo.lan*. A single e-mail from the malicious attacker then grows to *N* or more e-mails - all generated and transmitted by *bravo.lan*.

A combination of the three issues described above can lead to the following attack scenario.

4. Attack scenario

Assume *beta.lan* is a mail host serving a major organization with high bandwidth connectivity to the Internet. The organization employs a mail gateway server (*bravo-gateway*) which checks all incoming e-mails to *@bravo.lan* for viruses, then forwards them to an internal mail server (*bravo-internal*). The internal mail server is configured to generate NDN messages, which unfortunately combines all three faults presented previously. I.e. every NDN message includes a complete copy of the original message that is sent to the *mail from <address>* for each and every failed recipient address.

A malicious user on *alfa.lan* now sends *one* e-mail to *bravo.lan* having the following properties:

1. The e-mail has an attachment of *A* bytes.
2. The e-mail has *N* invalid Cc: recipients on *bravo.lan* (e-mails ending with *@bravo.lan*).
3. The reply-to address is faked to be *alice@delta.lan*.

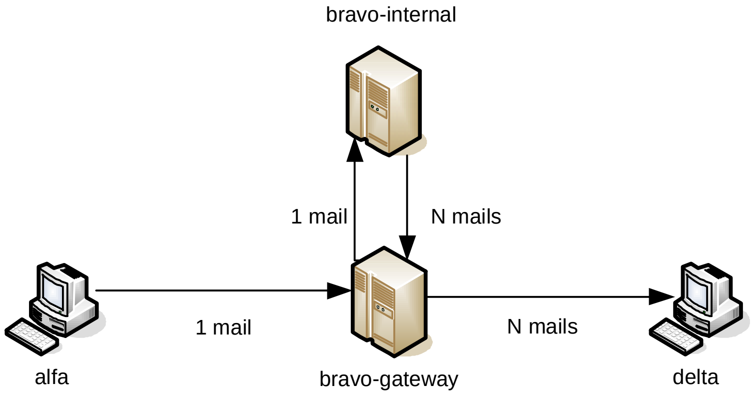

Fig. 2. A malicious user at *alfa.lan* sends a mail with *N* invalid recipients to *bravo.lan*, having spoofed the originator address to come from *delta.lan*. The *bravo-gateway* accepts the e-mail for all *N* recipients for the domain *bravo.lan* and forwards it to the internal mail server. The internal mail server is configured to create one NDN message for every invalid recipient, therefore creating *N* non-delivery messages to be sent to the spoofed originator, *delta.lan*.

The sequence of events is as follows:

1. *alfa.lan* sends the carefully crafted e-mail through its own SMTP server to *bravo.lan* (being the *bravo-gateway* server).
2. *bravo-gateway* accepts delivery of the mail for all recipients ending in *@bravo.lan*. It then forwards the mail to *bravo-internal* (after checking for viruses).
3. *bravo-internal* cannot deliver the mail to the *N* recipients, as they all are invalid. It therefore creates one NDN message for each of the *N* invalid recipient and sends them to the gateway for delivery to the originator of the e-mail, *alice@delta.lan*.
4. *bravo-gateway* sends those *N* NDN messages to *alice@delta.lan.*
5. *delta.lan* has to handle *N* incoming messages although it never sent any e-mail to *bravo.lan.*

The number of invalid recipients *N* determines the number of mails sent to the target at *delta.lan*. Through a combination of logic failures, a malicious attacker has gained a powerful force multiplier with respect to data volume and number of mails likely to be delivered. For *one* unit of data transmitted by the attacker the target gets inundated with *N* units of data - whereas *N* simply depicts the number of invalid *Cc:* or *BCc:* addresses the original e-mail was sent to.

The success of such an attack pattern depends on the availability of mail servers generating NDN messages as *bravo.lan*. An analysis of real SMTP mail services that manage e-mail for the worlds top organizations was carried out by the authors. The results of this study are described in the following section.

5. Experimental Verification

Initial manual testing of a small subset of public domain addresses revealed that some mail servers would respond with a large volume of NDN messages. The authors subsequently decided to proceed with an experiment designed to answer the following questions against a more representative set of mail servers:

1. How many mail servers accept invalid users at the initial SMTP session?

For the servers that accept any recipient name, we wished to know:

2. How are NDN messages generated?
3. How many NDN messages can be received in response for one mail sent?

5.1 Setup of Experiment

Our initial experiment (referred to as “Experiment A”) consisted of SMTP servers from a large number of randomly selected domains. The authors compiled a set of 12,451 domains to determine the number of hosts that do not reject invalid users at the initial SMTP session handshake. Our initial batch included six invalid recipients in every probe e-mail sent. For the subset of hosts responding to the discovery probes we ran two separate batches, each with 25 and 100 invalid recipients in the probe e-mails in order to analyze how NDN messages were generated. Finally, for a further subset of hosts, we ran a batch with 1’000 invalid recipients in an effort to ascertain whether some hosts had any reasonable limitations in place.

Table 2. Distribution of the top 10 *top-level domains* (TLD) in the list of 12’451 target hosts used. These TLD’s represent 96.8% of the hosts in our list. CH is the top level domain for Switzerland, from where the experiments were run.

TLD	Number	TLD	Number
com	61.1%	net	2.0%
ch	23.5%	uk	1.1%
gov	2.9%	edu	0.4%
org	2.8%	au	0.4%
mil	2.2%	de	0.4%

A supplemental experiment (referred to as “Experiment B”) was initiated to clarify mail relay issues with organizations that maintain more than one SMTP server. The experiment was configured similarly to the first but focused upon a different set of SMTP servers. This second experiment targeted a large number of the “Fortune 500” companies.

This second experimental set consists initially of 302 “Fortune 500” domains. For each server listed within their domain registration containing an MX record, a unique e-mail was sent. This email consisted of four (4) invalid recipients.

5.1.1 Anatomy of a probe mail

The probe mails were largely identical except for the *reply-to* address and the *number of invalid recipients N*. For each batch, the target hosts were positively identified by a unique *reply-to* address in the probe mail. These *reply-to* addresses pointed to individual mailboxes on our systems for the purposes of automatic collection, identification, analysis and correlation of the inbound e-mail.

1. One probe e-mail per target server and mail-batch
2. A constant number *N* of invalid recipients per mail and batch.
3. A plain text message body of 1,500 bytes.
4. An attachment (image) of 7,200 bytes.

5.2 Results Summary

The discovery batch of Experiment A targeted 12,451 hosts with one probe mail per host having *N*=6 invalid recipients each.

Table 3. Number of hosts and responses for the discovery-batch

Discovery-Batch (*N*=6)	Values	
Hosts targeted (= mail sent out)	12,451	100%
Hosts responding (1 or more NDN messages)	7,458	59.0%
E-mails received	9,158	73.5%
Bytes out	100.256 MB	100%
Byte in	119.825 MB	119.5%

An important observation is that 73.5% of the hosts respond with a NDN message instead of issuing a *550 no such user* response at the initial SMTP communication. These 9,158 hosts unnecessarily generate NDN messages of which 442 hosts (5.9%) generated more than one NDN message in response of a single e-mail.

Mail systems responding with multiple NDN’s were subject to closer examination with *N*=25 and 100. For a selection of 105 hosts responding to these probes we sent a batch with *N*=1,000 invalid recipients.

Table 4. Responses for e-mail batches with *N*=25, 100 and 1000 invalid recipients. Only one e-mail was sent out per target host and batch. Only responding hosts are used for the calculation.

Batch	25	100	1000
Hosts targeted (=mails out)	442	442	105
Hosts responding	401	367[1]	102
E-mails received	14,043	23,044	81,768
Bytes out[2]	3.42 MB	3.93 MB	3.60 MB
Bytes in	185.14 MB	317.56 MB	1,146.32 MB
Mail multiplier	31.77	52.13	778.75
Data volume multiplier	53.97	80.80	318.04

With 105 outbound e-mails totaling 3.60 MB of traffic we caused the mail servers under study to generate more than 80’000 e-mails, totaling 1.15 GB of traffic, within 6 hours after the e-mails were sent out. At this point in the experiment we reluctantly decided not to probe additional mail servers with 1,000 or greater invalid recipients due to bandwidth constraints.

Experiment B refined the testing techniques used in the initial experiment. Starting with a sample of 302 “Fortune 500” domains consisting of 730 unique SMTP servers (MX records found in the 302 domains) responses were received from 204 “Fortune 500” domains – corresponding to 414 responding STMP servers.

Table 5. Responses for the “Fortune 500” e-mail batch with *N*=4 invalid recipients.

Batch	Fortune 500
Hosts targeted (=mails out)	730
Hosts responding	414
E-mails received	801
Bytes out[3]	5.7 MB
Bytes in	10.4 MB

It is important to note that of the 302 “Fortune 500” domains sent to, 77% (232) of these had more than one MX record listed in their domain registration details. In addition, as some domains utilized shared mail services (such as anti-spam and anti-virus gateways) the 414 SMTP servers responding constitute 430 unique delivery-to SMTP hosts.

[1] Some hosts black-listed us for some time after the first batch completed.
[2] The volume of data to transmit a probe mail grows with the number of recipients.
[3] The volume of data to transmit a probe mail grows with the number of recipients.

Table 6. Analysis “Fortune 500” e-mail response by frequency.

Responding Hosts	Freq.	Total	Min	Max	Avg.	Mulitplier	Std.Dev
1 Response	301	4032046	899	23111	13396	x 1.7	2312
2 Responses	7	82433	4552	17500	11776	x 1.5	6759
3 Responses	3	131192	40365	50430	43731	x 5.5	5802
4 Responses	118	6146370	6168	67158	52088	x 6.6	8442
>4 Responses	1	66341	66341	66341	66341	x 8.4	0

5.3 Mail multiplier, data volume multiplier

The two most interesting numbers determined are the *mail multiplier* and the *data volume multiplier*, which are calculated for every batch of probe mails sent.

1. The *mail multiplier*, gives the multiplication factor in NDN messages the targeted host sent in response to one single probe mail it received.
2. The *data volume multiplier* is the multiplication factor in the number of bytes the targeted host sent back for each byte it received.

6 Analysis

Experiment A

The findings of Experiment A show that 59.0% of the 12,451 hosts respond to invalid recipients with one or more NDN messages, whereas 3.5% of the hosts (or 5.9% of the responsive hosts) reply with *more than one* NDN message. However, those few hosts have a high potential for being used as mail or data volume multipliers. In Figures 2 we plot the number of NDM messages received (x-axis) against the value of the data volume factor (y-axis) for the mail-batch with 100 invalid recipients.

The authors were astonished to find out that about half a dozen hosts probed continued to send NDN messages even weeks after the experiments were completed.

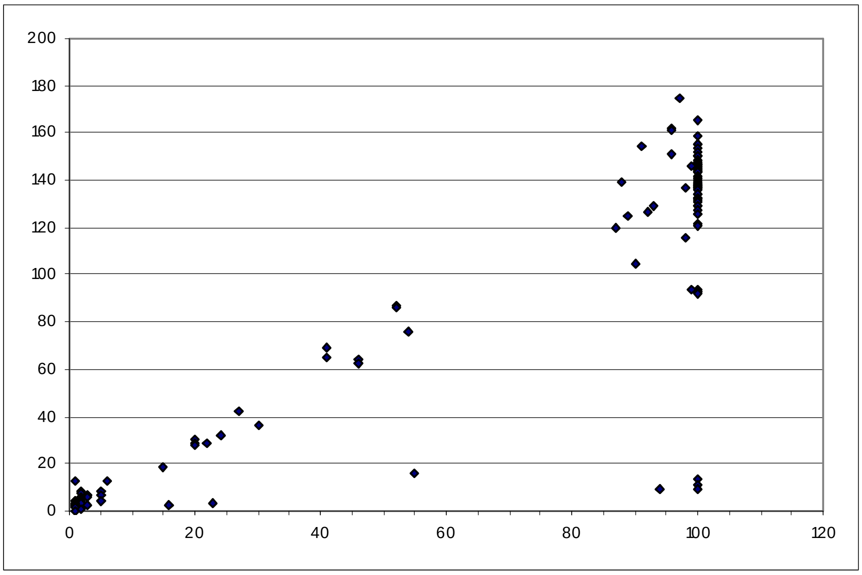

Fig. 3. Number of received NDN messages (x-axis) plotted against the data volume factor (y-axis) for each host responding to a probe-mail with 100 invalid recipients. The hosts located on the upper right generate the most e-mails and data volume when triggered by a single incoming e-mail. E.g. some hosts respond with 100 NDN messages sending 160 times the data volume of the incoming mail. Note: The graphs for the other mail batches appear similar and are currently available online [6].

Experiment B

The findings of Experiment B show us that by including secondary and backup SMTP servers, the probability of a SMTP server responding with more than one NDN message increases. Of the 430 unique responses from the 414 responding SMTP hosts, fully 30% of these hosts responded with more than one NDN for each e-mail initially sent.

Table 7. Responding host analysis for the “Fortune 500” e-mail batch with *N*=4 invalid recipients.

Responding Hosts	Frequency	Percentage
1 Response	301	70%
2 Responses	7	2%
3 Responses	3	1%
4 Responses	118	27%
>5 Responses	1	< 1%
1 Response Only	301	70%
More than 1 Reponse	129	30%

The distribution of NDN messages for Experiment B was similar to the first experiment, with the exception being the percentage of SMTP servers responding with more than one (1) NDN.

Table 8. E-mail responses received from "Fortune 500" SMTP servers.

E-Mail Size	Bytes	Multiplier
Sent e-mail	7900	
Smallest individual received e-mail	899	11%
Largest individual received e-mail	23111	293%
Average individual received email	13056	165%
Smallest cumulative received mail-box	899	11%
Largest cumulative received mail-box	67158	850%
Average cumulative received mail-box	24322	308%

Experiment B required the sending of an e-mail with four invalid recipients to each SMTP server defined by a MX record in their domain registration. Consequently, unlike the e-mails of Experiment A which included up to 1000 invalid recipients, the mail multipliers are much less.

However, any SMTP server responding with 4 or more NDN messages per sent e-mail is highly likely to respond with 100 or even 10,000's of NDN's if 100 or 10,000's of invalid recipients were included in the original sent e-mail. Therefore this multiplier would increase proportionally with the number of invalid recipients as per Experiment A. As it stands, the average size of responses to a single sent e-mail resulted in a 308% increase in NDN response size.

6.1 Interpretation

1. Most of the plotted responses appear far beyond a data volume multiplier of 1.0 (y-axis), which means these systems are ideal force multipliers - sending out many more bytes for one byte received.
2. The number of NDN messages received varies between one and the number of recipients *N* of the probe mail. A few hosts were found to return even more than *N* NDN messages.
3. For a given number of NDN messages received (x-axis), the data volume factor (y-axis) between different hosts can vary greatly. This is indicative of the variety in methods employed to generate NDN responses.

Tabulated summary data for each batch of probe-mails sent:

Table 9. Single worst host for the batches with *N*=25, 100 and 1000 invalid recipients.

Batch	25	100	1000
Hosts targeted	442	442	105
Response of worst host in batch			
Hosts	1	1	1
E-mails received	325	273	5,999
Bytes in	1.094 MB	2.623 MB	94.993 MB

6.1 Mail headers

The diversity in the size of the received NDN messages for identical probe e-mails (a factor off 36) can be explained as follows:

1. Some systems only send a small error message.
2. Some systems append the whole list of failed recipients.
3. Some systems append the complete list of failed recipients with a transcript of the SMTP session for each recipient.
4. Some systems route e-mails between multiple internal mail servers. Every hop adds information to a certain degree; at least a *received by* line in the header or more information as described above.
5. Some systems append parts or all of the original e-mail message body
6. Some systems append a complete copy of the original e-mail including all attachments

In many of the NDN messages received, the authors found internal information of the organization's infrastructure which is valuable for an attacker to find specific vulnerabilities and to fine tune an attack [3].

6.1 Potential for Denial of Service

The experimental data unambiguously shows that a high proportion of mail systems generate NDN messages in a way that can be misused and used to launch attacks as described in section 4. The authors discovered mail servers that appear to have no upper limit on the number of recipients within an e-mail. Flooding such a system with well-prepared e-mails has the potential to consume bandwidth and server resources to a point that the mail system will become unresponsive.

Many of the systems we found to be prone to such an attack belong to larger global organizations or governments, presumably having high capacity connections to the Internet. Causing one or more of these mail systems to mail-bomb a nominated target can also make the targeted system unresponsive.

By targeting non-primary SMTP servers, the probability of receiving more than one NDN response to a single e-mail increases. This is most likely due to a combination of SMTP relay rules and the inability of many mail systems to identify valid existing e-mail recipients.

Experiment B also highlighted the following:

1. Organizations that had chosen to utilize the services of external anti-spam and anti-virus organizations for the primary SMTP services were more likely to respond with *N* factor NDN message responses.
2. Even if the primary SMTP server is not configured to respond with more than one NDN message, targeting the organization through their secondary or backup SMTP services may initiate multiple NDN messages per *N* invalid recipients. In this experiment 7% more domains were found vulnerable to becoming NDN DoS agents through their secondary SMTP services.

6.1 Distribution of malicious content

Mail servers sending a copy of the original e-mail in the NDN message can be misused to send any content to any target by spoofing the originator address of the e-mail. This can lead to problems such as:

1. Who is responsible for the content sent out (if company X sends you a virus)
2. Social engineering (company X sends you a NDN message with an attachment, presumably from you)
3. Denial of service through filling the targets mailbox making it denying legitimate mails (mailbox saturation error)
4. Denial of service through mass mailing malicious content such as compression bombs [4] of any kind.

7 Recommendation

Unfortunately there are many mail servers that do not allow for direct configuration on how NDN messages are generated. However there are many control mechanisms available that could help prevent misuse of mail systems.

7.1 Do not accept mail for invalid recipients

The single best method to prevent this kind of abuse is to make every publicly available mail server aware of what valid users may be served through it. It is imperative that invalid users are rejected upon the initial SMTP session handshake, which then relieves the receiving mail server from generating any NDN messages. Mail border gateways and anti-virus/anti-spam gateways are especially prone to accept any recipient for the Internet domains they serve. Many mail servers accept e-mail to any recipient for the domains they serve in order to prevent spammers to test for existing e-mail accounts. However, especially in the light of the findings presented here we consider this a bad approach.

7.2 Limit the maximum number of recipients

There should be an upper limit on the maximum number of recipients a mail server will accept in a session. Attempts to send e-mails to more than the allowed number of recipients should be refused by a *452 Too many recipients* response at the SMTP session. In RFC-2821 section 4.5.3.1 a lower limit than 100 recipients is discouraged for unknown reasons [5]. Whatever the reasons for a higher limit were, the authors believe it is imperative to have a limit and that mails to invalid recipients are not accepted.

7.3 Generate few error messages

To prevent the misuse of the system as an e-mail multiplier, the server should generate no more than one NDN message for every e-mail received. Ideally the server does not respond instantly but collects the information per originator and sends out a condensed NDN message after a period of time. Any mechanism that automatically generates e-mails after being triggered by external events must be carefully designed. Such automatic responses tend to worsen the situation by generating more traffic and processor-load during malicious activities.

7.4 Generate small error messages

The NDN message generated has to be as small as possible. It is unnecessary and, in the authors opinions, dangerous to include a copy of the original e-mail in the NDN message. Further more, it does not make sense to include transcripts of internal SMTP communications between servers in the NDN message. Such data usually leaks internal information about the infrastructure to the outside, a valuable source for any hacker [3]. The authors believe that the failed recipient address and the first 100 characters of the subject are more than sufficient information to notify the sender.

7.5 Input data validation

In general, it is best practice to thoroughly validate all user data submitted in to an application. In the present case, mail servers represent the application and any communication has to be properly validated.

8 Conclusion

Through a series of experiments, the authors have been able to identify a significant flaw in the way e-mails are managed by SMTP services globally. Through poor NDN message design, a considerable proportion of mail services currently deployed throughout the Internet may be used as denial of service agents. By abusing a small number of vulnerable mail servers within large organizations with high Internet bandwidth connectivity, it is possible to cause the complete denial of service of critical e-mail services of any targeted organization.

The current configuration and design processes of secondary or out-sourced SMTP mail services increase the number of viable domains that can be used as DoS agents.

It is a simple process of abusing multiple SMTP services to cause a Distributed DoS (DDoS) that would increase the impact on the target. Given the possibilities with payload multiplication factors, should an organization host their main SMTP services in-house, network bandwidth saturation is also possible – causing a DoS of all Internet connectivity.

Analysis of a large number of representative SMTP servers throughout the Internet leads the authors to believe that this vulnerability is endemic and requires very little technical skill to instigate. Organizations should review their SMTP server configurations to ensure that they cannot be used to DoS agents, and have a planned response plan should they become targeted as an SMTP DDoS victim.

The authors have been similarly astonished to find out that a number probed hosts continue to send multiple NDN messages weeks after the experiments were completed.

References

1. RFC-821 - Simple Mail Transfer Protocol - J. Postel. Aug-01-1982 - www.ietf.org/rfc.html
2. RFC-2821 - Simple Mail Transfer Protocol - J. Klensin, Ed.. April 2001 - www.ietf.org/rfc.html
3. Gunter Ollmann, Passive Information Gathering - http://www.technicalinfo.net/papers/PassiveInfoPart1.html
4. Compression bombs - http://www.aerasec.de/security/advisories/decompression-bomb-vulnerability.html
5. Detailed data of the experiments - http://www.techzoom.net/mailbomb