An NCC Group Publication

Automated enumeration of email filtering solutions

Prepared by:
Ben Williams

Contents

1 List of Figures and Tables

2 Introduction

This paper summarises research undertaken in 2013-2014 to develop offensive reconnaissance techniques for automated and external enumeration of the email filtering solutions of target organisations. We show how a methodology, automated scripts, and test message sets can be used to enumerate a target email filtering solution, quickly and to a high level of detail and accuracy. Enumeration described here is performed without requiring any exploits but using product and service features which are there by design.

Details which can be enumerated by an external attacker include:

- The email filtering managed services, software, or appliance products, often with version information, hostnames, and internal IP addresses.
- A detailed picture of the filtering policy in place, including identification of policy or configuration loop-holes.
- The capability of the products and services in use, and their ability to handle identification of hidden threats in more challenging formats (such as embedded within various documents, archives or other specially chosen attachment format-types, specially encoded message formats, or deliberately corrupted messages).

In many cases the above information can be gained without interaction with internal users, and with a low likelihood of detection. Information on filtering weaknesses identified using these techniques could be used by attackers in a reconnaissance phase, to help tailor targeted phishing, malware, or client-side attacks, by providing information on effective ways to bypass the specific filtering in place.

This paper builds on previous work by the author in identifying vulnerabilities in a variety of security appliances[1] and gateways[2], and shows how the discovery and further enumeration of vulnerable email security solutions can be achieved, even when these systems are not directly externally exposed.

In some situations this type of enumeration can be fast and straightforward; in other situations it can be more complex, and we will discuss how some obstacles can be overcome.

We also take some time to explore easily constructed malicious attachments which evade most filters, and how these can be combined with basic social engineering to gain remote code execution on typical corporate desktop environments.

The techniques described in this paper were developed in a variety of situations including penetration testing engagements, product capability assessments, and sampling tests on a variety of leading organisations.

[1] Hacking Appliances: Ironic exploits in security products
https://media.blackhat.com/eu-13/briefings/B_Williams/bh-eu-13-hacking-appliances-bwilliams-wp.pdf

[2] They ought to know better: Exploiting Security Gateways via their Web Interfaces
https://www.nccgroup.com/media/18475/exploiting_security_gateways_via_their_web_interfaces.pdf

3 An Overview of Common Email Filtering Solutions

Here we give a brief overview of typical topologies, options, and practices for email filtering.

3.1 Common Email Filtering Solutions and Topology

A brief overview of typical email filtering systems and characteristics is provided in this section, to aid understanding of subsequent sections. Various options are available for email filtering, and examples of the following topologies were seen during NCC Group's research and tests on a variety of organisations.

3.1.1 Email Managed Services

One of the most common solutions for email filtering is to use an external email managed service. In these cases the MX records for the target domain point to the third party. Emails are processed and filtered externally and forwarded to the target organisation's internal mail server or boundary mail server. This requires that the internal email server is accessible via an external IP address, so it is important that effective access control is implemented to prevent external parties from connecting directly to the mail server (bypassing the filter). This ensures that all inbound mail is processed by the filtering service. Optionally, outbound filtering can be achieved by routing outbound messages via the managed service.

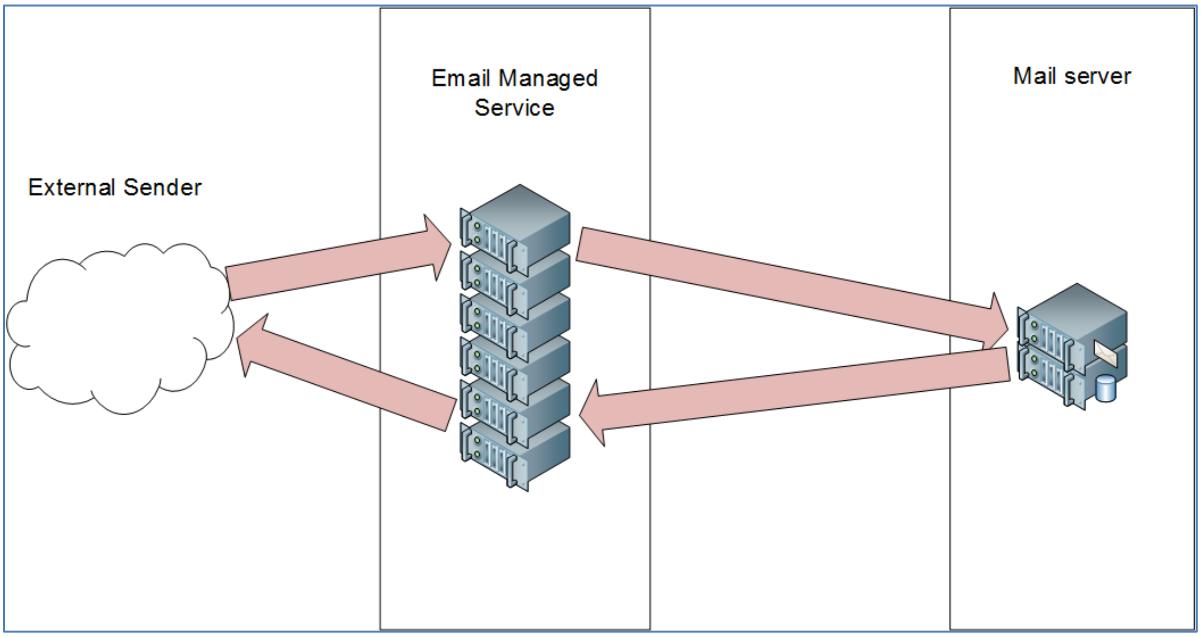


Figure 1 Email managed service only

Typically, managed service email filtering offers the benefit of scale. These services process very large volumes of spam and malware for many domains, collating statistical data across many systems, and tracking sender IP reputations accurately based on a statistical profile of previous messages sent from each known sender IP address. This can give an excellent level of protection in terms of blocking untargeted threats. However, these services are typically limited in terms of their ability to perform deep content analysis (finding embedded executable code in document and archive attachments for example) and are often limited in terms of defining complex granular policy rules for groups or individual users.

3.1.2 Onsite Email Appliance (or Software Filtering Solution)

Another popular choice is to use an email filtering appliance (or multi-function gateway) onsite at the target organisation. In this case the MX records point directly to these hosts for email filtering.

As well as offering good protection for spam and malware, some appliances can be configured with complex customised policies and can perform improved detection of embedded threats (as well as typically offering a more complete email security feature set than a managed service). Additionally, the organisation can retain improved confidentiality, as the messages are not routed and processed via a third party.

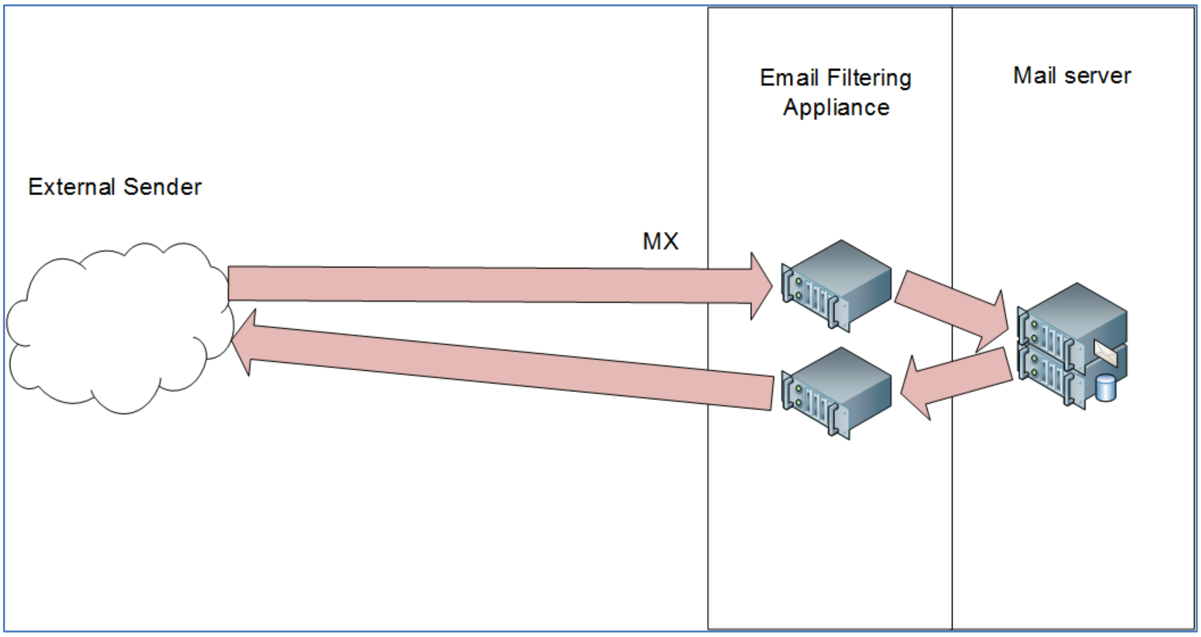


Figure 2 Onsite appliance only

3.1.3 Multilayer Filtering

When compared, email filtering managed service and appliance solutions each have strengths and weaknesses, and there is benefit in having multiple layers of filtering. A common configuration is to have two layers of filtering; the first being a managed service, providing bulk sanitisation for generic spam, phishing, and malware attacks. The second layer would typically be onsite email filtering appliances, which may perform more filtering, with deep content analysis, more complex logical rules, and refined granular policies for specific groups of users or individuals. Additionally, different anti-malware solutions from different vendors can be used in different layers, to increase the effectiveness of filtering for known malware.

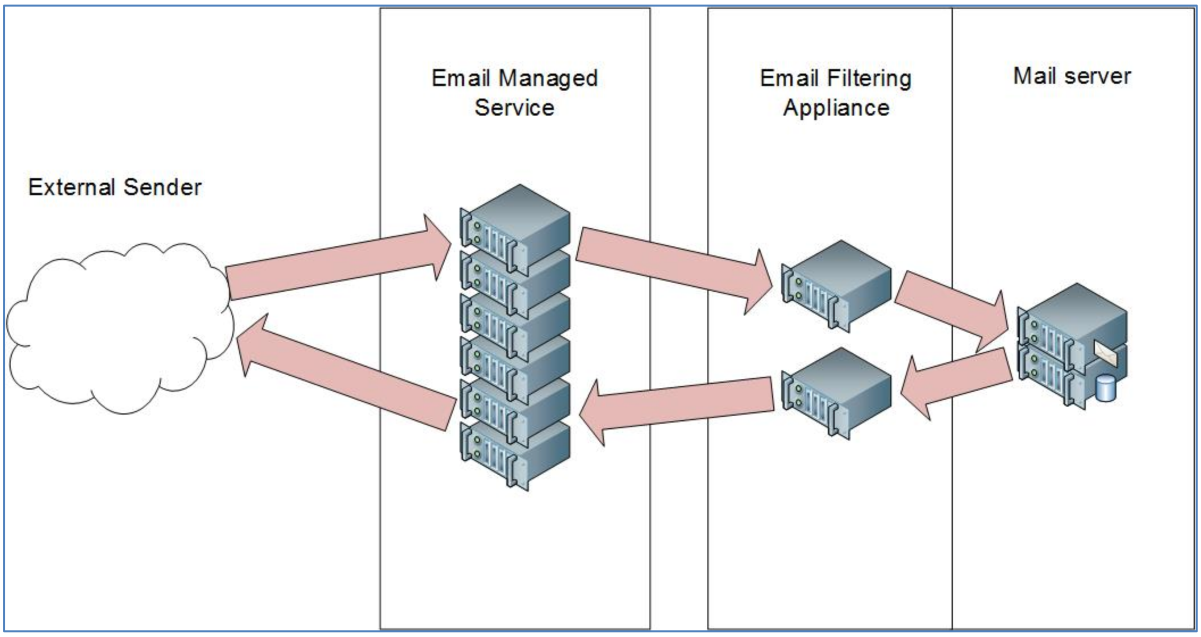


Figure 3 Multi-layer filtering

In tests on leading organisations who were using a multilayer approach, it was seen to be more often the case that where two layers were used, the managed service was only used inbound, and outbound messages were typically delivered directly from site to the recipient's MX record using the email filtering appliance.

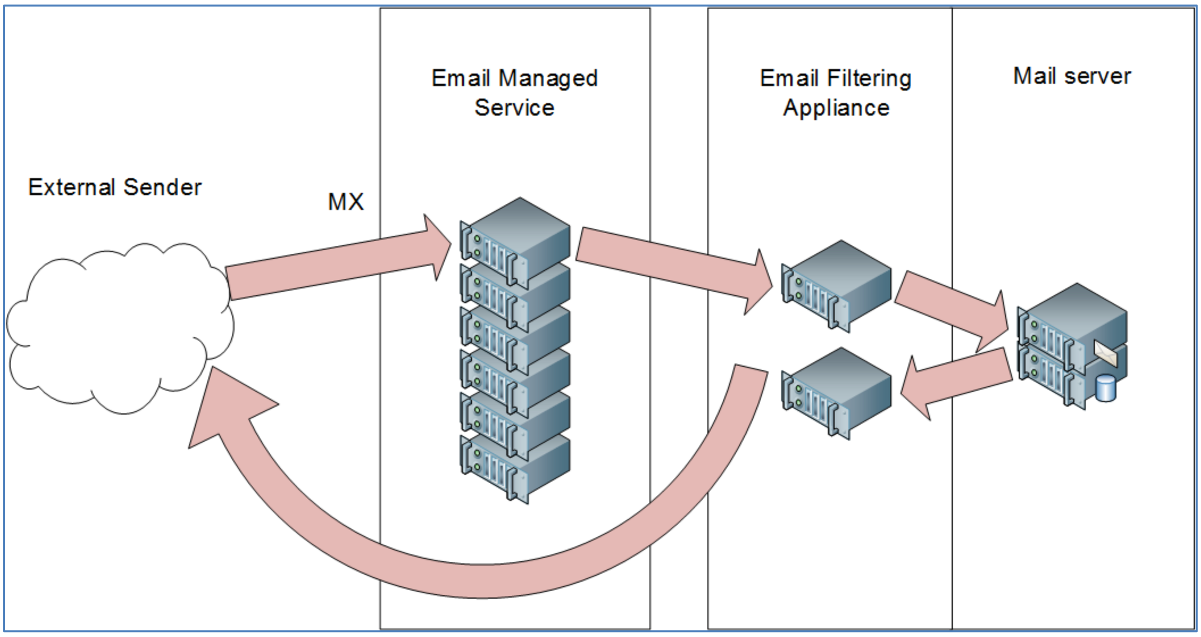


Figure 4 Multi-layer filtering is typically not used for outbound email

3.1.4 Other Options

There are a wide variety of products and services available for email filtering, with offerings from many vendors. Though the above cases were observed in the majority of leading organisations tested, a small number of cases were seen to have solutions which did not fit the topologies described above. Other options included fully hosted email services, or using filtering software which runs as part of the internal email server itself. Some deployments were seen to have multiple managed services and multiple appliances from different suppliers, deployed in series, though this level of complexity was rarely seen, with one or two filtering layers being the norm (accounting for over 97 per cent of cases observed).

3.2 Presenting a Consistent Defence

Whilst any of the options previously described may be valid depending on the risk acceptance profile and IT security budget of an organisation, a consistent defence should be presented to external attackers. All inbound email routes should have the same security posture. For the MX records presented, each record should direct email to a system with the same capability and filtering policy (ideally the same product or service), otherwise an attacker could choose to send their messages via systems with weaker filtering.

During this research it was observed that some organisations had MX records which were not configured to be consistent, or there were other topology weaknesses. For example, where a managed service was used for some of the MX records, while other MX records pointed to a different managed service, or pointed directly to email security appliances which were located in the layer behind the managed service.

A small number of instances were seen where one or more of the MX records pointed to an SMTP relay acting as a simple store and forward service, before forwarding messages to a managed service or appliance. This can have a negative effect on some filtering technologies, such as IP reputation-based services, as connections coming from a generic forwarding service may result in a different score than those coming directly from systems known to be sending malware and spam.

3.3 Access Control Issues

For access control to be effective, it should not be possible to trivially bypass any layer of inbound filtering by sending emails directly to the next layer of filtering or the internal mail server. In tests on leading organisations NCC Group were able to show that typically between five per cent and ten per cent of organisations tested were vulnerable to either full or partial bypass of their filtering solution for inbound email. These issues may not be obvious unless a thorough audit of the email systems is conducted, but techniques for quickly finding access control-related filter bypasses are described in the next section.

3.4 Filtering Policy Best Practices

Spam, phishing, and known malware filtering are primary filtering goals for inbound email, but to protect users, other threats such as unknown executable code, scripts, documents containing macros, or malicious links should be filtered. It is also important to manage encrypted attachments appropriately, as these could contain any of the threats listed above. These threats should be detected, and action taken, such as quarantining potential threats and optionally alerting administrators or the intended recipient.

3.4.1 Deep Content Inspection

Executable code and scripts should be identified by mime-type, file extension, and file signature. Additionally, email filtering products and services should unpack encoded attachments, including popular types of documents, archives, and compound files, and identify threats contained within them, and this unpacking should be performed in a multi-layered approach. The reason for this is clear; most modern email client software hides or disables attached executables and scripts to prevent users from running them directly by double-clicking. However, archives and documents do not have the same level of restriction (though some restrictions are enforced by modern versions of Microsoft Office).

Malware infections such as the Cryptolocker[3] ransomware are typically sent as email attachments in documents or archives (for example as an executable file in a zip file, with a double file extension so as to appear to be a PDF rather than an EXE, in default Windows desktop configurations).

When products and services unpack encoded content in multiple layers (deep content inspection) this increased functionality can detect more threats in email, but it should also be noted that this extra analysis can introduce additional risks such as denial of service, or remote code execution via memory corruption vulnerabilities in the product performing the analysis.

3.4.2 Limiting Information Disclosure

Limiting information exposure regarding which security products are in use is desirable to help defend against targeted attacks against the filtering products in use. Preventing external attackers from enumerating the filtering policy in place is also important, as in any implemented filtering solution; there are normally filtering weaknesses which may be exploited if known, to deliver threats to internal users.

[3] US-CERT Cryptolocker information
http://www.us-cert.gov/ncas/alerts/TA13-309A

4 Enumeration Techniques

The following section describes techniques for enumeration of email filtering solutions, gives examples of information disclosed, and shows some statistics collated from sample tests of an automated tool (MailFEET) used in a variety of scenarios.

Tests were mainly performed in three scenarios:

- Targeting specific products in a controlled test environment, sending hundreds of payloads to identify product capabilities and weaknesses.
- Detailed analysis during customer engagements, including penetration tests, phishing and client-side attacks, and email filter policy reviews.
    - These included active attacks against real users
- Limited payloads and tests of a wide variety of domains, with a special focus on leading organisations.
    - These tests were targeted for a large number of domains, but with a small number of messages per domain
    - Executable but inert proof-of-concept payloads were sent to non-existent recipients.
    - This set of tests was mainly used for collecting data for producing discovery signatures, and for gathering statistics on typical implemented policies.

4.1 Managed Service Enumeration

One of the quickest and least intrusive enumerations which can be performed is to determine email filtering managed services in use. In most cases this can be done simply with DNS and “whois”.
For example, for the domain “example.com”:

```
# dig example.com MX +short
30 example.com.s5b1.psmtp.com.
40 example.com.s5b2.psmtp.com.
10 example.com.s5a1.psmtp.com.
20 example.com.s5a2.psmtp.com.
```

The domain portion of the hostnames returned can be matched against a list of known managed service domains. Here, it is evident that the Google Postini managed service is in use (also note that the records are consistent as they all point to the same service, which is good).

A small percentage of organisations using managed services use their own domain for the hostnames of their MX records, and point these to IP addresses of the managed service. Though this is unusual (around two per cent), it means it is slightly more accurate to resolve the MX record hostnames to IP addresses, and use “whois” to resolve the owner of the IP addresses, to determine managed services. For example:

```
# dig mail1.example.com +short
64.18.4.10

# whois 64.18.4.10 | grep OrgName
OrgName:         Postini, Inc.
```

The above can easily be automated to quickly and accurately determine the managed services used by large numbers of domains. There is not really a way to effectively obscure the use of a managed service, as these systems generally need to be the first hop in order to benefit from the managed service’s connection-based filtering. This does not present an issue in itself, unless an attacker is aware of specific filtering weaknesses associated with a particular managed service. Performing this test across the Fortune 500 provided the following information:

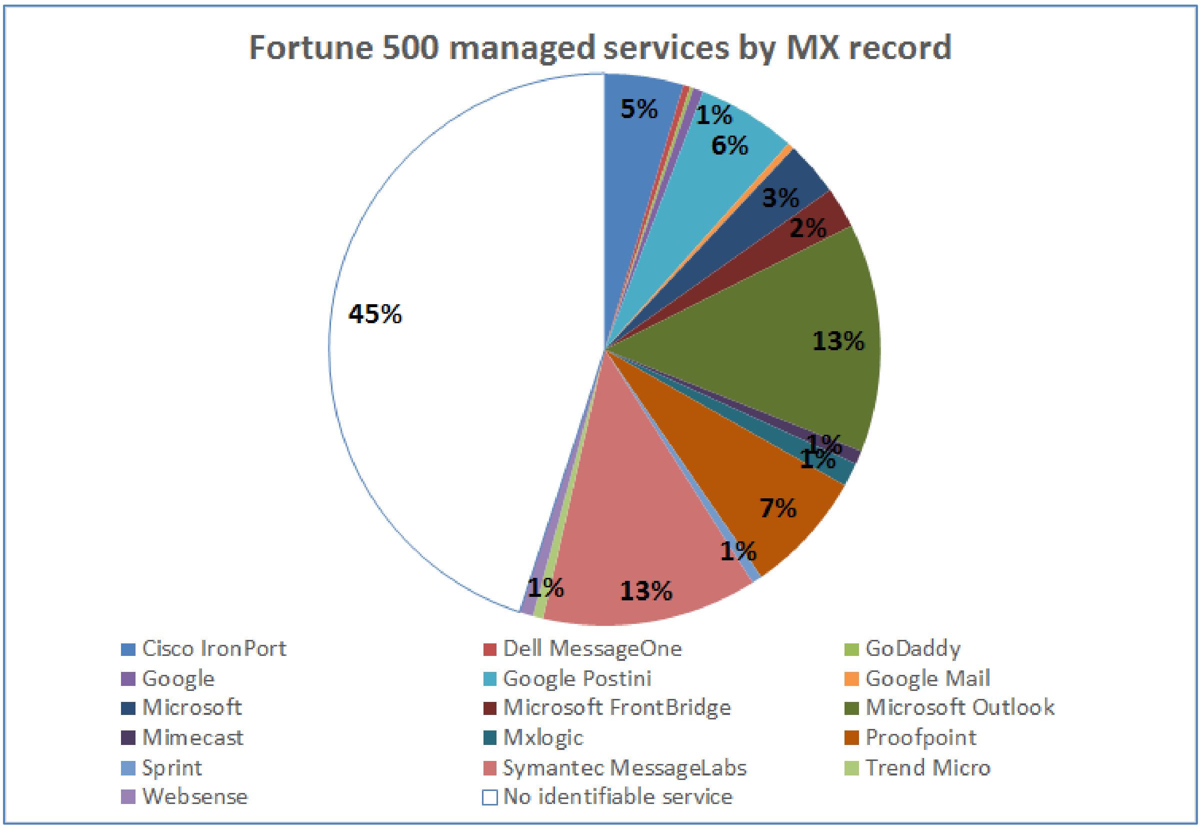


Figure 5 In tests, 55 per cent of the Fortune 500 were seen to use known managed services

Figure 5 shows that the leading email filtering managed services for the fortune 500 were: various Microsoft services (eighteen per cent), Symantec MessageLabs (thirteen per cent), Proofpoint (seven per cent), various Google services (seven per cent), and Cisco IronPort managed services (five per cent).

Forty-five per cent of the Fortune 500 did not appear to use managed services for email filtering, but in further testing various filtering appliances and software products were identified using other techniques.

In any case, it is normally clear which managed service is in use from SMTP conversations (such as the banner or other text in responses) or from X-headers, received headers, or other message annotations, using message analysis as described below. For these reasons the author considers it to be impractical to try to obfuscate which email managed service is in use for a given organisation.

4.2 Appliance Product Identification With Port-Scanning and Banner Grabbing

Standard vulnerability scanning techniques include port-scanning, banner grabbing, SMTP verb enumeration, and parsing HTTP/HTTPS web UIs or HTTPS certificate information. These could be used to identify product and version information where services are directly exposed to the Internet. These techniques are covered by existing tools and methodologies, were not required during this research, and will not be covered in this paper.

Whilst it is true that product version information can typically also be found by connecting to the default port of the administrative web UIs for many popular email security products, best practice is to restrict access to these interfaces. Additionally, the above techniques cannot be used to discover products hidden behind other SMTP services, or located in a DMZ with strictly limited access control, with few or no ports externally exposed, and default banners changed.

4.3 Product Enumeration by Message Analysis

The main method of enumeration discussed in this paper is message analysis of replies, obtained by sending messages in order to receive responses from internal systems. This can be achieved for example by sending messages to non-existing recipients, or to other auto responders, or gaining email responses from real users (though techniques in this paper focus on automated and non-interactive techniques).

In tests, NCC Group observed that it is standard practice for most email security products and services to mark all messages processed with received headers and x-headers, and sometimes to annotate the message body with information which could be useful to an attacker in a reconnaissance phase.

Information disclosed in this way included product vendors and versions, internal IP addresses and hostnames, update versions, and processing scores or outcomes for various types of spam and malware filtering.

```
(Sentrion-MTA-4.3.1/Sentrion-MTA-4.3.1)
```

```
(Symantec Messaging Gateway) with SMTP id
```

```
(TREND IMSS SMTP Service 7.1)
```

```
MailMarshal (v7,1,0,4874)
```

Figure 6 Examples of product versions, from “Received” headers and X-Headers

```
Remote-MTA: dns; [172.30.46.17]
```

Figure 7 Internal IP addresses or hostnames: errors in message body

```
Received: from xxxxxx.xxxxxxx.xxx (10.173.160.32) by
xxxxxx.xxxxxxx.xxx (10.173.160.241) with Microsoft SMTP Server
```

Figure 8 Internal IP addresses and hostnames: received headers

While this level of information disclosure is intended for troubleshooting purposes, and is normally considered a low-severity risk (unlikely to cause problems in isolation), it should be noted that this information is accessible to external attackers, so it is best practice to minimise the disclosure, especially where there is disclosure of exact product versions and internal addresses. Additionally, when you consider that most email filtering appliances tested in previous research were found to have serious vulnerabilities[4] [5], this ability to enumerate product versions has an increased level of associated threat.

```
X-IronPort-AV: E=Sophos;i="4.93,874,1378875600";
```

```
X-IronPort-AV: E=McAfee;i="5400,1158,7286"; a="160098426"
```

```
X-Proofpoint-Virus-Version: vendor=nai engine=5400 definitions=5800
signatures=585085
```

```
X-Proofpoint-Virus-Version: vendor=fsecure
engine=2.50.10432:5.11.87,1.0.14,0.0.0000 definitions=2013-12-12_01:2013-12-
11,2013-12-12,1970-01-01 signatures=0
```

Figure 9 Examples of different antivirus plugin options, used with some products and services

Some products and services give a choice as to which antivirus engine is used, offering multiple options. Knowing which option is used (and the update version) could be useful to an attacker targeting end-users with malware or client-side attacks, because the attacker could more accurately confirm that their payloads would not be detected in advance of the attack.

```
X-esp: ESP<57>=
       SHA:<0>
       SHA_FLAGS:<0>
       UHA:<10>
       ISC:<0>
       BAYES:<31>
       SenderID:<-1>
       DKIM:<0>
       TS:<17>
       DSC:<0>
       TRU_spam1: <0>
       TRU_profanity_spam: <0>
       TRU_money_spam: <0>
       TRU_medical_spam: <0>
       TRU_urllinks: <0>
       URL Real-Time Signatures: <0>
       TRU_lotto_spam: <0>
       TRU_watch_spam: <0>
       TRU_scam_spam: <0>
       TRU_freehosting: <0>
       TRU_legal_spam: <0>
       TRU_adult_spam: <0>
       TRU_stock_spam: <0>
       TRU_playsites: <0>
       TRU_phish_spam: <0>
```

[4] Hacking Appliances: Ironic exploits in security products
https://media.blackhat.com/eu-13/briefings/B_Williams/bh-eu-13-hacking-appliances-bwilliams-wp.pdf

[5] They ought to know better: Exploiting Security Gateways via their Web Interfaces
https://www.nccgroup.com/media/18475/exploiting_security_gateways_via_their_web_interfaces.pdf

```
X-NAI-Spam-Score: 1.5
```

```
X-Proofpoint-Spam-Details: rule=notspam policy=default score=41 spamscore=0
ndrscore=41 suspectscore=3 adjustscore=0 phishscore=0 adultscore=0 bulkscore=0
classifier=spam adjust=0 reason=mlx scancount=1 engine=7.0.1-1305240000
definitions=main-1308150307
```

```
X-SpamScore: 2
```

Figure 10 Examples of spam scores (useful feedback for avoiding spam filtering).

Spam scores can be enumerated; although this does not have an overly negative impact on security, in tests it was clear that it was possible to modify messages with the aim of reducing spam scores to improve delivery rates. Adding business-like word content, legal disclaimers, and office documents significantly lowered the spam score. This can be monitored interactively, and messages adjusted accordingly, to maximise the likelihood of messages being delivered.

Though several email security products have the capability to remove previously added “Received” headers from email they process (to try to reduce information disclosure), this practice would appear to be rare. In tests on a sample of 152 organisations it was possible to enumerate some email security products and services in use in all of them, mostly by analysing message headers.

4.4 Product Type and Version Enumeration With Probe Messages

By creating signatures for the various information disclosures, and sending test messages to non-existing users at 152 leading organisations, it was possible to enumerate a wide variety of products in use.

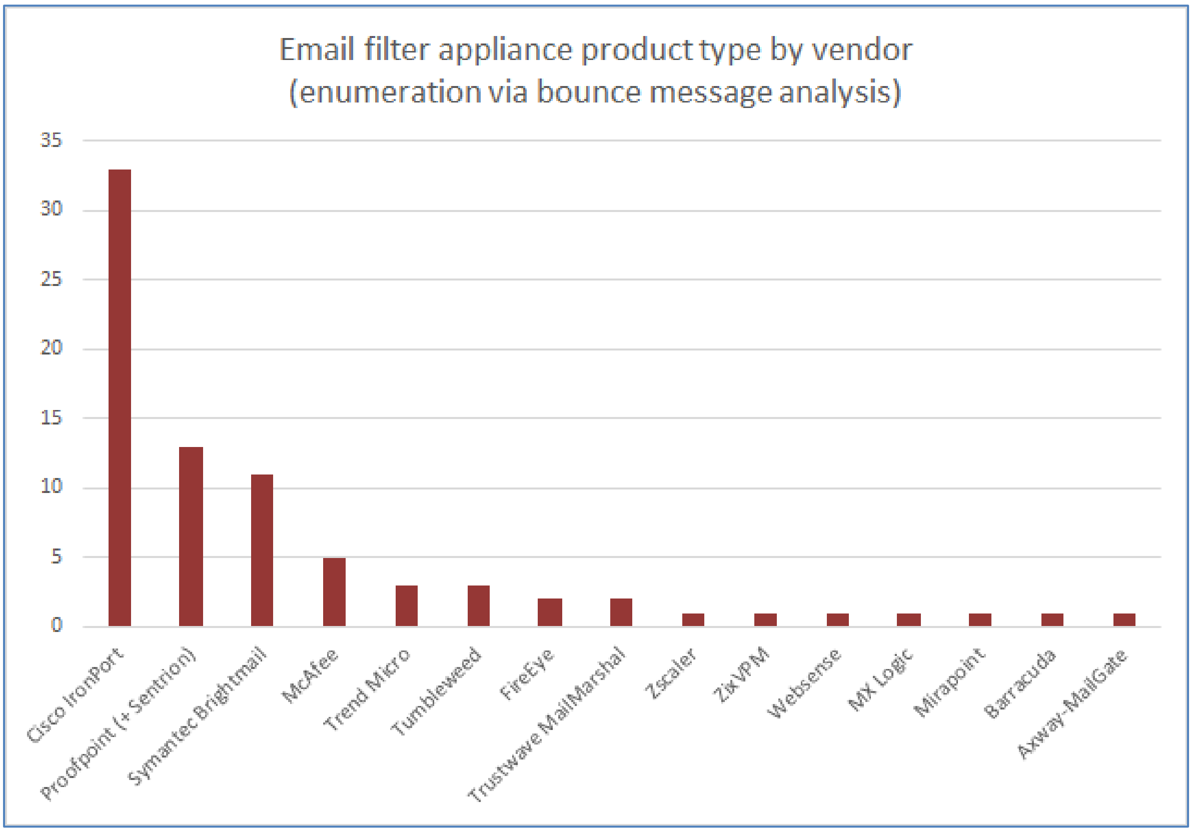


Figure 11 Appliance and software product enumeration by vendor for domains of 152 leading organisations

The above data is from message analysis of data in bounce messages, but interestingly the most popular products and vendors found are similar in some respects to the Gartner magic quadrant and associated analysis of “Secure E-Mail Gateways” [6]. Though Gartner’s data collection and analysis methods are different, it is interesting to see how closely these results compare, in terms of highlighting popular products. [7]

Enumerated managed services and product statistics can be combined to see how many organisations have which type of solution, or multiple solutions. An analysis of this is shown in the diagram below; however, it is likely that some internal or onsite filtering solutions may have been missed, therefore the number of organisations using multiple filtering solutions is likely to be larger than the twenty per cent stated below.

[6] Note: Microsoft TMG was not considered, as the focus of this research was on enumerating managed services, appliance products, and policies.

[7] Gartner Magic Quadrant for Secure Email Gateways
https://www.gartner.com/doc/2538216

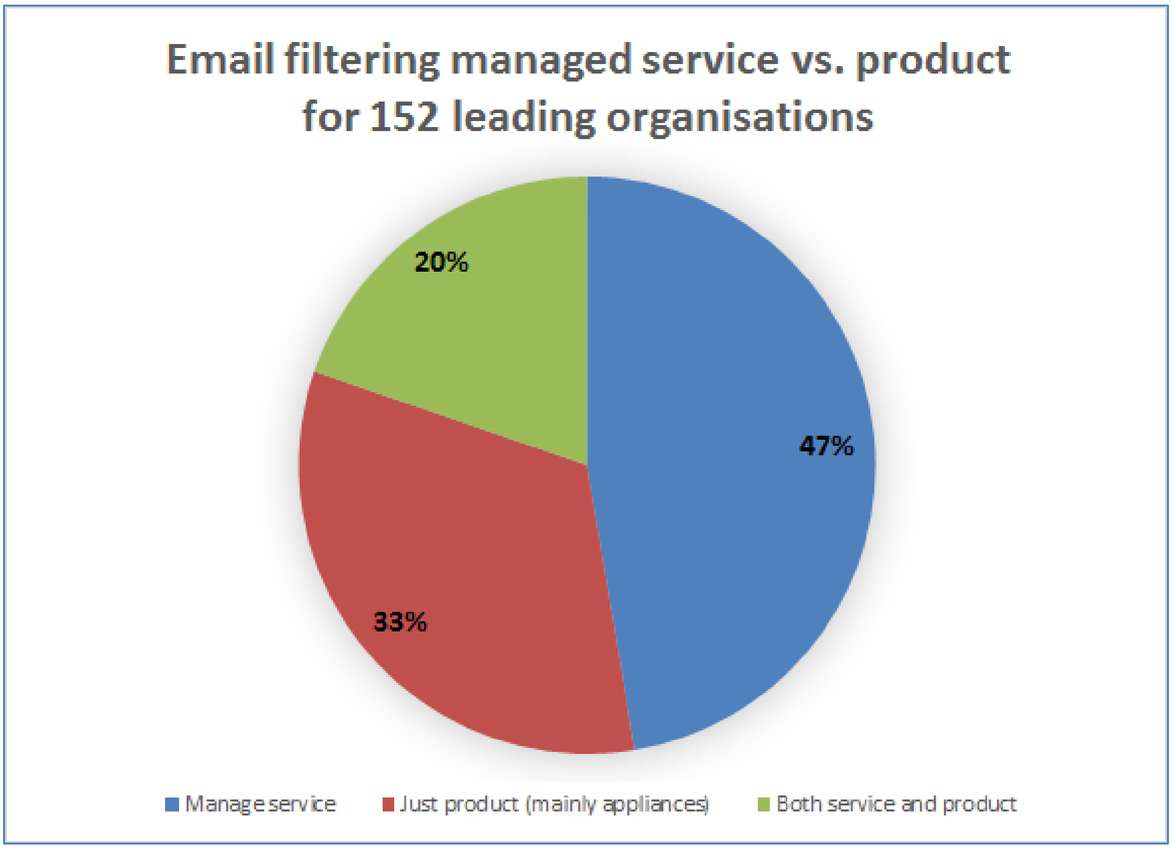


Figure 12 Appliance and software product enumeration for domains of 152 leading organisations

4.5 Bidirectional Disclosure in Bounce Messages

By default, when a message cannot be delivered because the intended recipient does not exist, most mail servers send a non-delivery report containing a copy of the original message and attachments to the original sender. When messages are bounced in this way this usually results in the two sets of headers going back to the original external sender. There will be headers on the non-delivery report message, showing the outbound path of the message. There will also be headers in the original message attached, showing the inbound path of the message. Different information can be revealed in these two sets of headers, as different paths can be taken depending on direction.

In received headers, typically a system reports on itself and on the system with which it is communicating, so different address information is disclosed for the same system, depending on which direction the message is going (internal IP address or hostname versus external IP address or hostname, for example).

4.6 Filter Bypass Enumeration With Probe Messages

Filter bypasses due to weak access controls can be enumerated in an automated way. This can be achieved by sending a series of probe messages to each of the MX records, and parsing the received bounce messages. Multiple messages are required, as load-balancing and failover sometimes mean that messages can take different paths. Next, regular expressions can be used on the received bounced messages, to extract all the external IP addresses and hostnames from the message headers and message body.

The external IP address of any system connecting to return a message is added to this list, and attempts are made to send messages directly to each of the new external IP addresses discovered (ones which were not part of the original MX records). Finally, delivery results and any new bounced messages are analysed to identify bypasses. Potential filter bypasses can be confirmed by checking that product version headers previously seen are no longer visible, showing that the messages have not been processed by the filtering product or service.

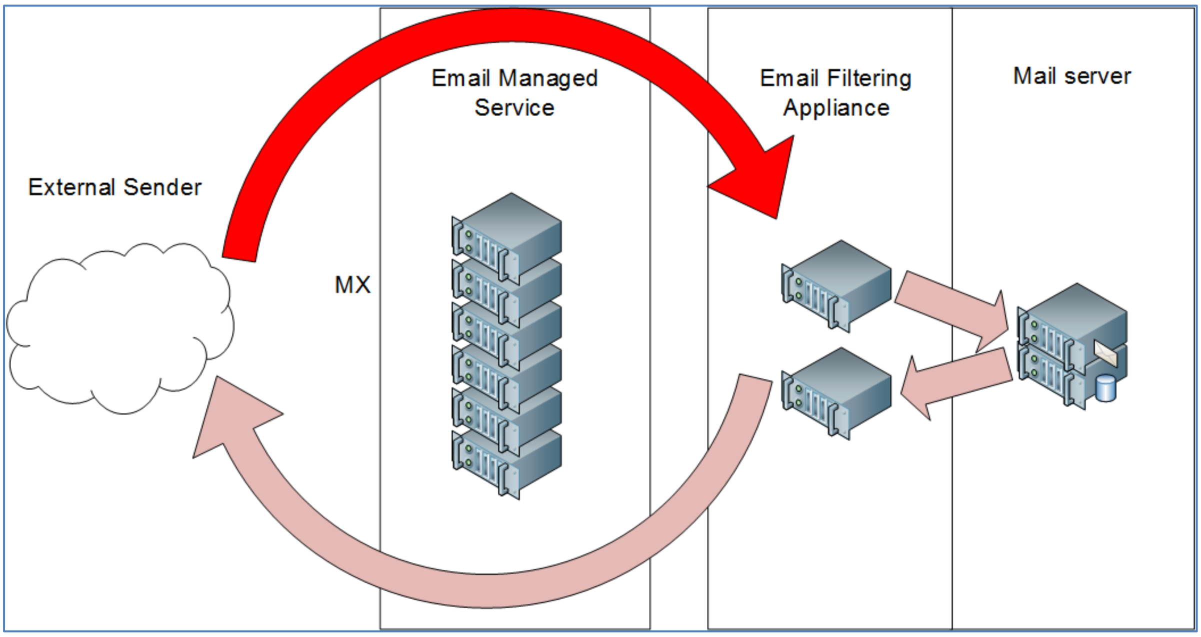


Figure 13 Direct bypass of one layer of filtering

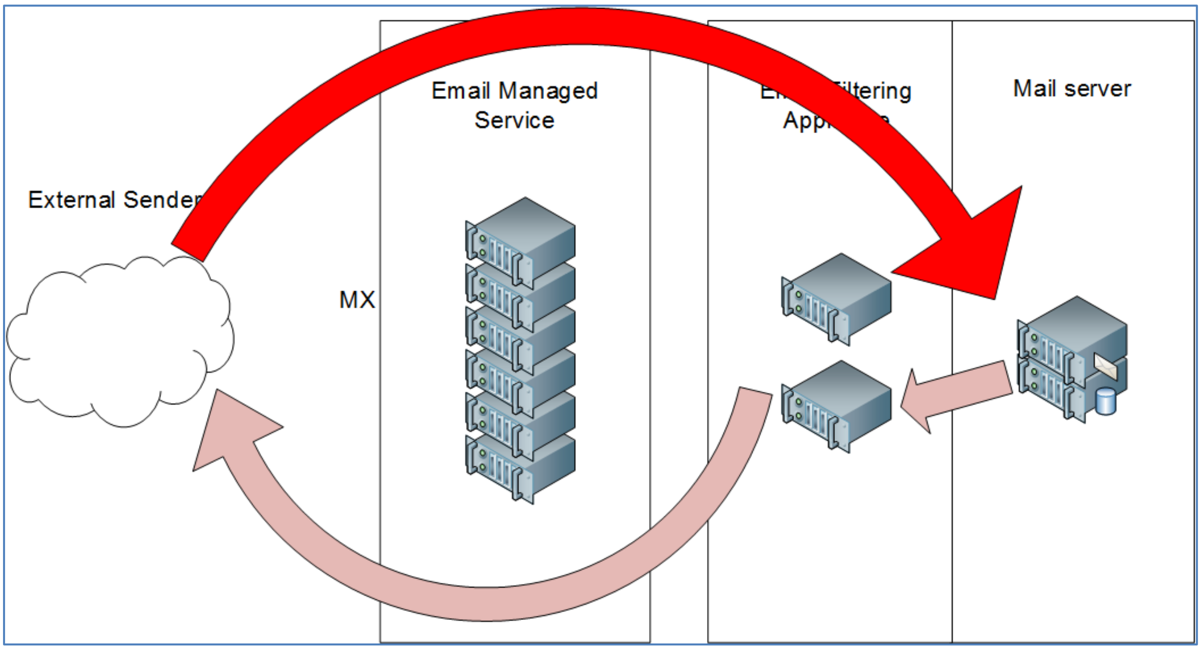


Figure 14 Direct bypass of multiple layers of filtering

4.7 Policy Enumeration With Probe Messages

By sending a series of test messages, some innocuous and some with potential threats attached, differences in response can be analysed to automatically enumerate the implemented policy.

4.7.1 Policy Enumeration With Non-Delivery Reports

If messages to non-existent recipients are accepted and bounced by an internal mail server (or other internal gateway) this results in the easiest and most reliable path to policy enumeration. Analysis of the Fortune 500 showed that nearly a third of these organisations (153) both accepted and responded to messages sent to non-existent recipients.

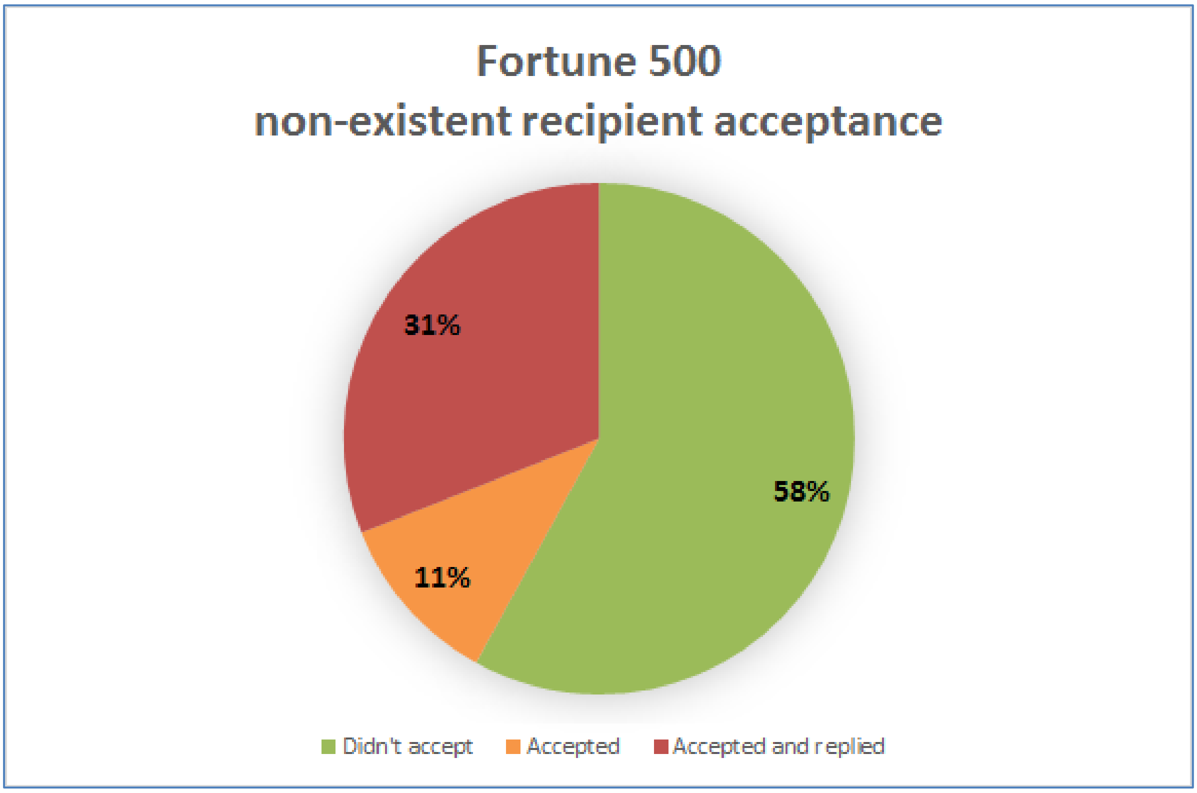


Figure 15 In tests forty-two per cent of the Fortune 500 accepted messages to non-existent recipients, thirty-one per cent generated automated replies

This meant that for a third of these organisations, policy enumeration was relatively straight forward. For the remainder, policy enumeration would likely still be possible, by abusing other types of auto-responder or by using specially crafted messages which generate processing errors.

By sending and tracking many messages with different attachments or encoding formats, it is possible to enumerate policy in an automated or semi-automated way. The default for popular email servers such as Exchange and Domino is to send a non-delivery report when a recipient does not exist, and by default this has attached the original message and its attachments. This means that delivery to the mail server can be confirmed by checking that the returned attachment is the same.

When a variety of messages are sent to a target domain, there are three main outcomes to consider:

1) No message is returned
    a. Normally one should assume that the original message has been filtered at some point along the path (usually inbound).
2) The original message is returned
    a. More often than not this means that the original message was not filtered, but more investigation is required to confirm this.
        i. If the original message is returned unmodified (with the same attachments), and the message was returned from the internal mail server or preceding relay, then one can assume it has not been filtered.
        ii. If the message is modified, this may mean that the original threat has been stripped, modified, or replaced – more investigation is required, but it is usually possible to analyse what actions occurred, and what this means in relation to policy.
3) A different message is returned, this could be:
    a. A policy block informational message to the sender – meaning the message would have been blocked
    b. A notification to the intended recipient, which was subsequently bounced by the internal mail server because the user did not exist – meaning the message would have been blocked
    c. Some other kind of processing failure message – which could have various causes and meanings (and should be assessed further)

Some of the possible scenarios are shown in the diagrams below:

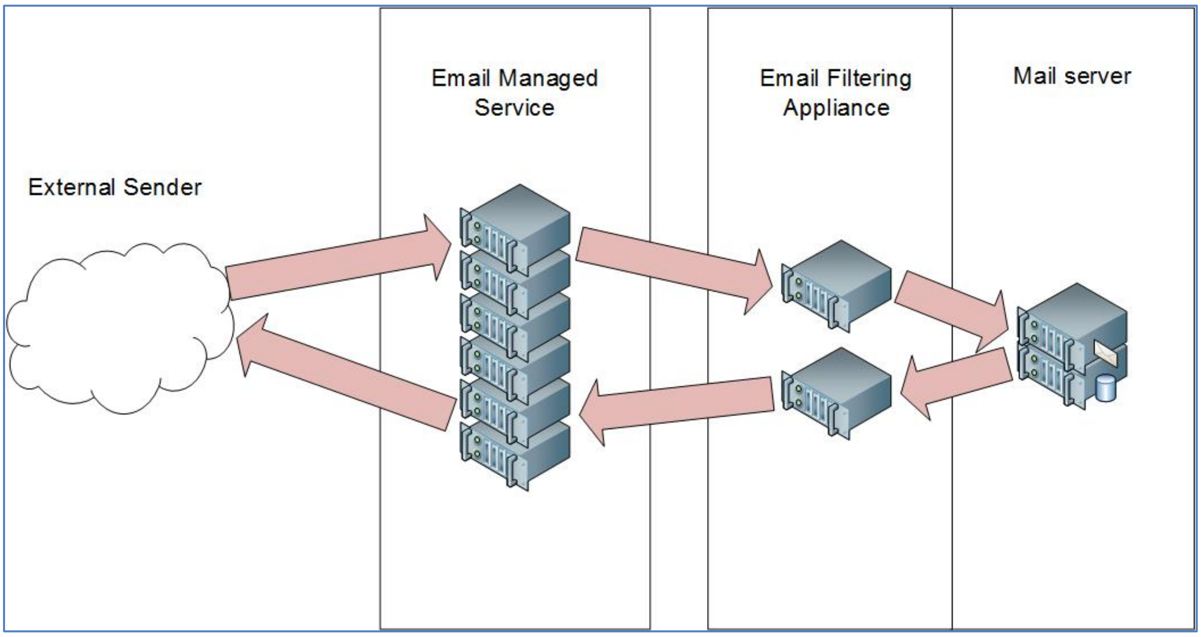


Figure 16 Bounce via non-delivery report, usually the sender gets the original message attached

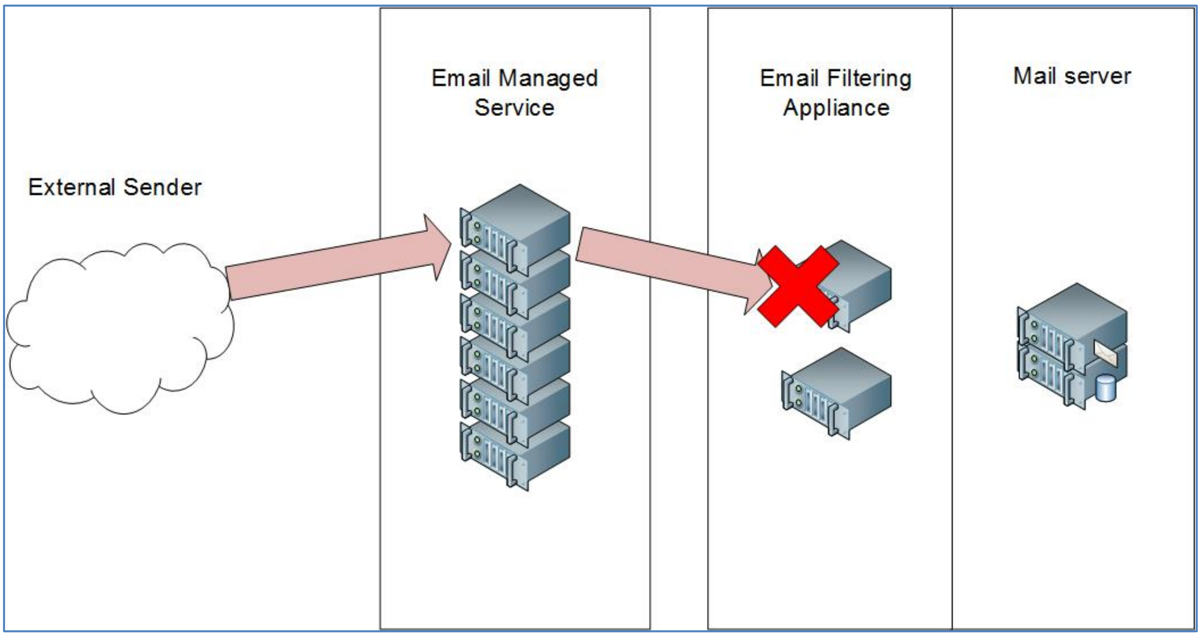


Figure 17 Message filtered therefore no response received

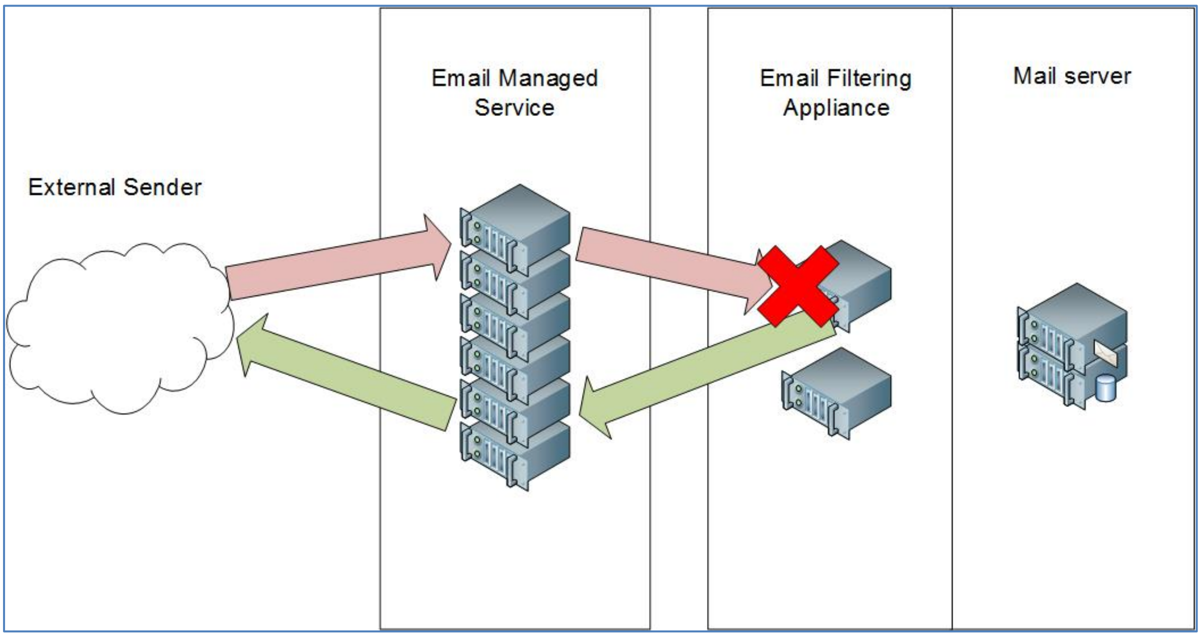


Figure 18 Policy block message generated

4.7.2 Informational Messages to Internal Users, and Modified Messages

Sometimes when a message is blocked or quarantined, an informational message is sent to the original intended recipient. If this message is spoofed on behalf of the original sender, or the original message is modified (replacing attachments for example), this can result in a bounce back to the sender containing further information. The filtering solution sends the modified message to the mail server, but as the user does not exist, this situation results in the informational message intended for the internal user going back to the original external sender.

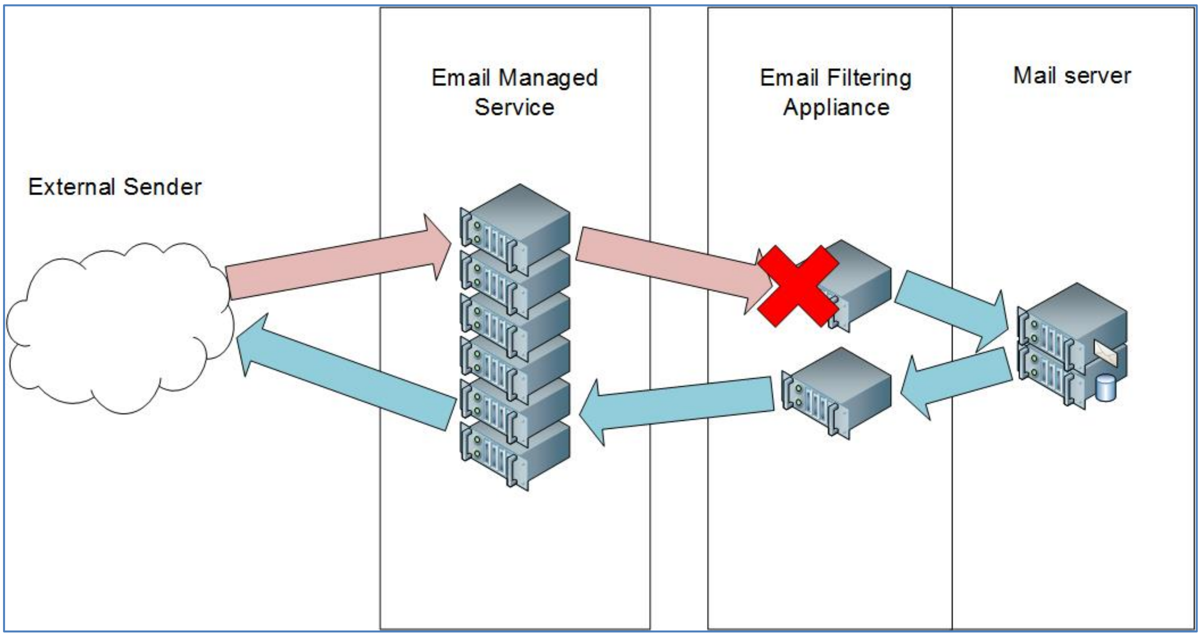
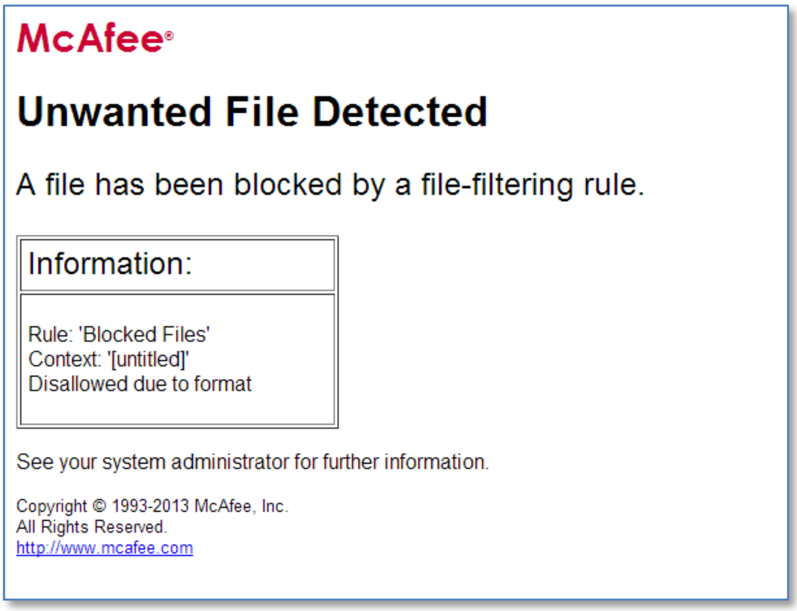


Figure 19 Information message intended for the recipient gets bounced back to the sender

During testing, several instances were found where messages were modified to remove potential threats. This can make policy enumeration a little more complex, but can also reveal more information about the products used, their capability and implemented policy. Modifications seen included:

- Attachment stripping (four per cent)
- Attachment replacement (two per cent)
- Attachment modification (one point five per cent)

Attachment replacement or modification can result in additional information disclosure, as shown in the following example, where an attachment with an embedded executable was removed, and replaced with a HTML file (intended for the internal recipient). This showed the product in use and policy outcome to the external sender:



McAfee®

Unwanted File Detected

A file has been blocked by a file-filtering rule.

Information:

Rule: 'Blocked Files'
Context: '[untitled]'
Disallowed due to format

See your system administrator for further information.

Copyright © 1993-2013 McAfee, Inc.
All Rights Reserved.
http://www.mcafee.com

Figure 20 In this case the attachment in the message was replaced by a HTML file which ended up going back to the sender. This can lead to more policy and product disclosure.

In rare cases, deep content message modifications were observed, such as the removal of executable code which had been embedded in Microsoft Office documents. This type of modification does present some additional risk, and generally speaking it is better to quarantine a message containing a threat rather than try to automatically fix the threat, and forward a modified message to the user.

4.7.3 Full versus Partial Enumeration of Policy

If a message is accepted then this usually means that some policy enumeration is possible, even if the message does not get all the way to the internal mail server, but is bounced by a preceding system. This is because, if a message is accepted, it will be processed against the policy before delivery is attempted, meaning that, at the very least, the policy on the service or product accepting the message can be enumerated.

If multiple filtering layers are employed, then how much of the policy can be enumerated depends on where the message is validated against a list of existing users. For example, if the username is validated by the receiving service of the second layer, then it may not be possible to enumerate the policy on the second layer, just the policy of the first layer.

4.7.4 Enumeration With Out-of-Office Auto-Responders

Clearly some domains do not accept messages to non-existent recipients (58 per cent of leading companies assessed did not accept messages to non-existent addresses) so an alternative solution is required for enumeration in these cases. One such solution is to use other types of auto-responders for addresses which exist, and one example of this is to use out-of-office messages for employees who are on leave or have left an organisation.

Though using out-of-office messages can be useful for enumerating products, services, and topology (as only a small number of test messages are typically required for this) there are some distinct disadvantages with abusing out-of-office replies for policy enumeration, as the test messages would typically end up in a real inbox, which would be annoying for that user. However, it may be possible to enumerate users who have recently left the organisation, or those on long-term leave, via social media sources.

The other main disadvantage for using out-of-office is that the bounced messages typically do not contain any content from the original message, therefore it is usually not easy to tell if the original inbound message had been modified (unless the subject line or message headers disclose this).

Advantages and disadvantages of using non-delivery reports as opposed to out-of-office messages include:

Non-delivery reports for non-existing users (around a third of domains)

- Non-intrusive
- Potentially quicker where grey-listing is implemented
- No need to find a real address with out-of-office or other auto-responders
- Enumerates the sum of inbound/outbound policy
- May cause blacklisting based on a threshold

Out-of-office (typically around ten per cent of users have it on, which varies seasonally)

- Requires real user reconnaissance in advance (LinkedIn etc.)
- Much more intrusive – as the message ends up in a real inbox
- Full-path enumeration to inbox
- Inbound policy only
- Higher delivery success rate
- Gains outbound message header info only

There is a small advantage in using both out-of-office and non-existent user bounces in combination. By combining OOA and NDR techniques this enables inbound/outbound policy differences to be determined, where outbound policy is stronger, for example:

- Where protectively marked documents are filtered outbound
- Where messages containing the word “confidential” are filtered outbound
- Where encrypted attachments are filtered outbound

4.7.5 Automation Versus Manual Review

When sending messages to a large number of domains, or sending a large number of messages to a single domain, automation becomes much more important. It is still important to have an oversight and manual review, as in some situations analysis can be challenging to automate programmatically in advance.

The best way to speed up manual review is to present key criteria indicators to the tool user, to help differentiate the various possible message processing outcomes. Key questions for messages are:

- Which messages had no reply?
- Which are bounces from the mail-server?
- Which have the original message?
- Which have the original attachment? (And is it the same?)
- Which are “block notification” messages either to the sender or intended recipient?
- Which are “other” types of messages (which required more investigation)?

Criteria seen to be important factors in this decision making process included:

- Comparing checksums of attachments received with those originally sent.
- Overall message size, and size of the original message if returned.
- Number of received headers
- Specific X-headers
- Message structure and attachments

Identifying specific text strings can also help with quick differentiation of message processing outcomes (though this varies heavily between organisations).

4.7.6 What Typically Gets Through?

In a variety of tests, focus was given to try to identify the simplest way to embed executable code or scripts, which allowed the transmission of an attacker's code, while being able to execute on the widest variety of corporate desktop environments and bypassing the majority of filters. Inert proof-of-concept pop-up executable payloads were used in unsolicited testing. In tests for NCC Group customers, either pop-ups or reverse shells were used. Typically the best formats from an attacker's perspective were found to be common office documents and zip archives, which most office workers could open and run any embedded code, macros, or scripts.

Multiple levels of encoding can assist in hiding executable content from simple filters (for example embedding an executable in a document in a zip file) though too many layers or complexity can raise suspicion with users.

In most cases it was found that simply embedding executables in documents, or sending password protected documents, evaded the majority of filters and resulted in delivery to the target mailbox.

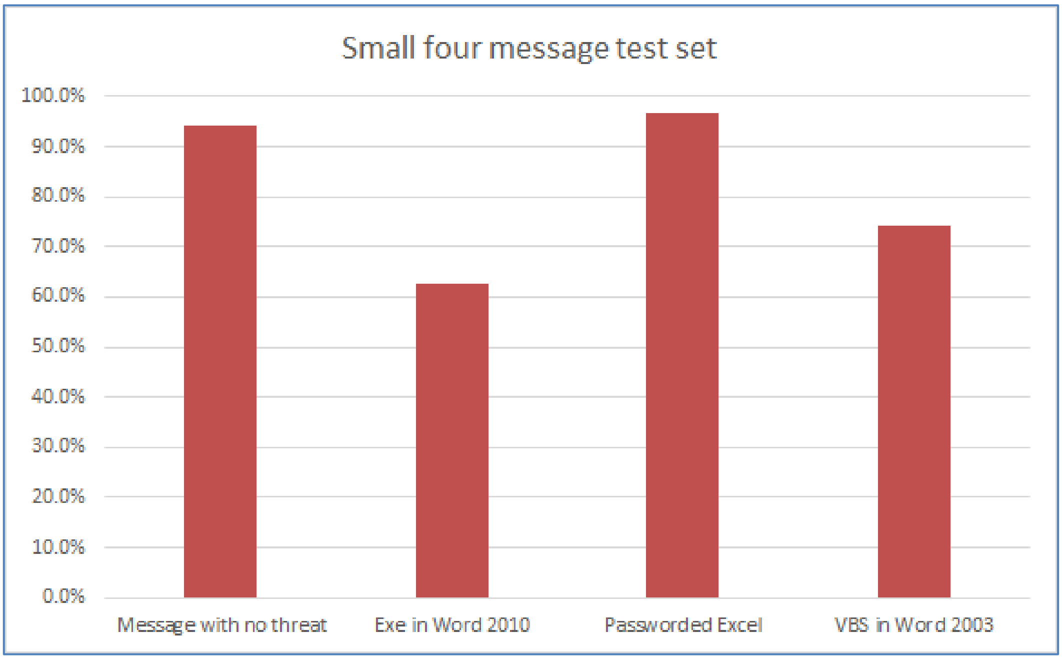
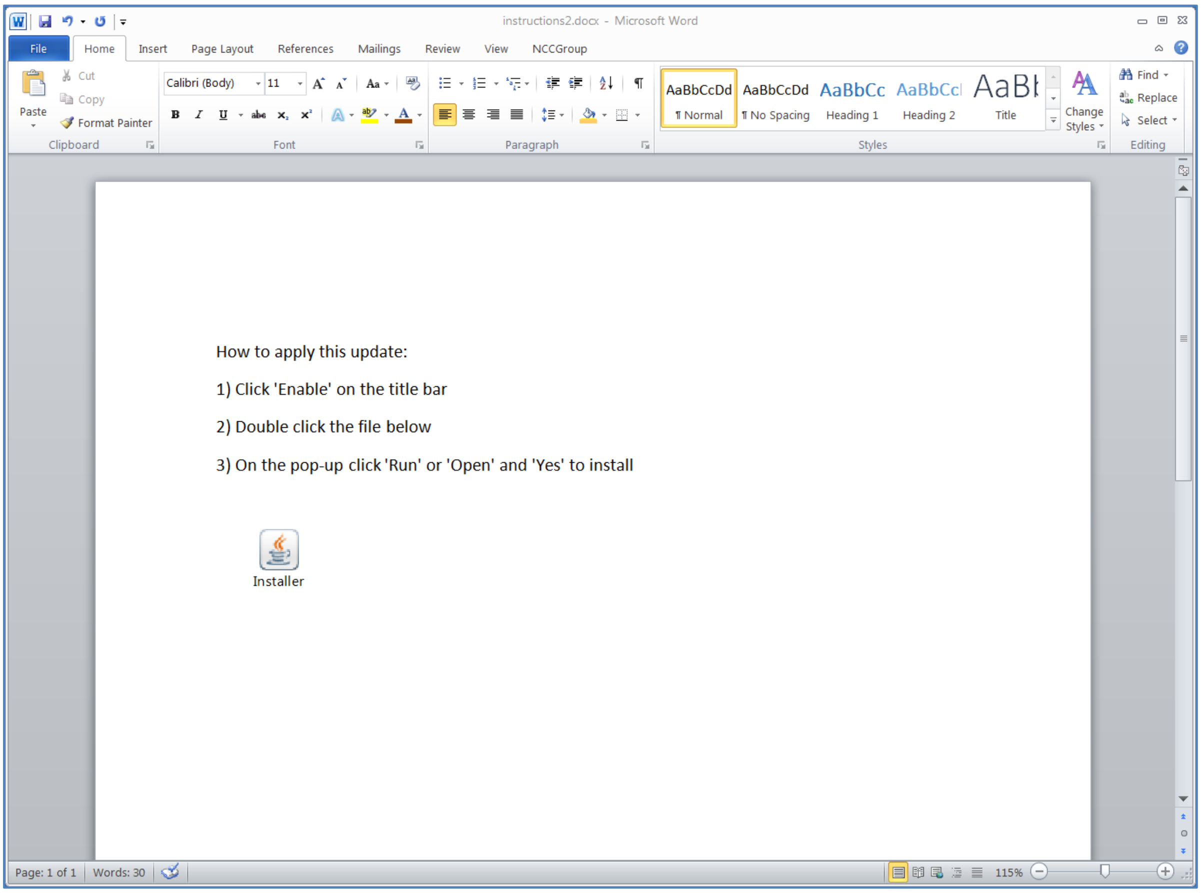


Figure 21 Here is a sample message set of four messages which were sent to 152 domains, and the resulting delivery rates for each

A clear difference was noted in the reasons for lack of detection, between various ways of embedding threats. For password protected documents, most products could detect these but organisations chose to allow password protected documents for business reasons. For executables and scripts embedded in unencrypted documents and archives, organisations generally would prefer to block these known threats, though some products and services were unable to detect them.

4.7.7 Attack Scenario Example

The following type of document has been used by NCC Group in client-side attack simulations as a follow-up to a phishing attack. Initially, users were sent spoofed messages from IT Support, asking them to login to a portal. The portal was fake and passwords were gathered (though login was not possible). An error message was presented on the portal stating the users' systems were missing a Java update. Users responded to the spoofed email complaining about the error message, and a follow-up document containing a set of instructions was sent:



How to apply this update:

1) Click 'Enable' on the title bar

2) Double click the file below

3) On the pop-up click 'Run' or 'Open' and 'Yes' to install

Figure 22 Client-side attack document sent to users, containing a "patch" and instructions

In these tests with NCC Group customers, the executable payloads used were either custom binaries, or standard Metasploit Meterpreter[8] shells (for example a reverse HTTPS shell) encoded with the Veil Framework Anti-virus Evasion[9] and embedded in documents with modified icons. These techniques resulted in successful code execution without detection.

[8] Metasploit Framework – Penetration testing and exploit development tools. http://www.rapid7.com/products/metasploit/

[9] Veil-Evasion is a tool to generate payload executables that bypass common antivirus solutions. https://www.veil-framework.com/framework/veil-evasion/

5 Further Research

5.1 Drive-by Enumeration of Web Filtering Solutions

Web filtering policies can also be enumerated externally in an automated way. Combined with the techniques described in this paper, enumerating web filtering solutions would give an external attacker a clearer picture of the filtering solutions in place.

Enumeration of web filtering solutions can be achieved by using JavaScript, or other client-side code in a “drive-by” scenario. For example, a JavaScript proxy could be used to force the internal client to request a series of test files from the attackers system, via the internal secure web proxy. This would not result in files being presented to the user, but filtering policy could be determined by the JavaScript, and a report sent back to the attacker’s system detailing which downloads were successful and which were not.

Further ongoing research and another paper “Drive-by enumeration of organisational web filtering solutions” will explore what can be achieved in terms of enumerating secure web proxy solutions.

6 Conclusion

In this paper we have discussed how an external attacker could enumerate products, services, and policies used for email filtering, and shown how policy bypass misconfigurations or product capability shortfalls can be identified.

Though the root causes of most of the issues discussed would typically be considered low severity information disclosure issues in isolation, in combination a variety of different disclosure issues can lead to an external attacker having the ability to build a clear picture of a target organisation's filtering solution in advance of an attack.

6.1 Recommendations for Email Security Implementers

This research shows that reducing information disclosure and improving both policy and configuration can greatly reduce exposure to potential threats from targeted client-side attacks. Various defensive configurations can be implemented to reduce the likelihood and impact of enumeration.

6.1.1 Do Not Accept Messages to Non-Existent Users

One of the easiest and most accurate enumeration methods relies on the target email domain accepting messages to any recipient, and the internal mail server bouncing messages sent on non-existent recipients.

As described above, this misconfiguration is surprisingly common, but can be corrected by rejecting messages to non-existent recipients at the boundary. A rejection should be in the form of a 5xx error message, performed at the first systems, those resolved by the MX records of the target domain. Usually a 554 error message is returned[10] as this indicates that the message is unwanted (without indicating specifically that the recipient does not exist).

Any rejection due to non-existing recipients also requires that this status of the intended recipient is known at the first hop, which requires an accurate list of real users' email addresses at the perimeter. For managed services this requires synchronisation from the user list in the internal directory or mail server to the third party providing the managed service. This can be done via a periodic update as a flat file, if organisations do not want to provide any external access to their internal directory server via LDAP. When accepting known, and rejecting unknown, addresses, it is very important to throttle and blacklist incorrect guesses to prevent brute-force user enumeration; most managed service email providers can do an excellent job of this.

It is likely that this "accept all inbound messages" approach is common because organisations have a concern of the sender not being notified when they make a legitimate error typing an email address. This concern is usually unfounded because for legitimate email, when an organisation rejects messages at the boundary, the preceding relay (usually the relay of the sender's own organisation) will generate a non-delivery report back to the sender. This non-delivery report message would not have any sensitive headers from the target domain, and so should reduce the ability of external attackers to enumerate products and policy.

[10] SMTP Error codes
http://www.serversmtp.com/en/smtp-error

Other implications of accepting messages to any recipients can result in the exposure of an organisation to increased cost and risk from processing unwanted spam messages. Also, by replying with the original message attached, this can be used as a form of open relay (where the attacker spoofs the intended recipient, to get messages delivered via a third party). This can lead to spoofing and phishing attacks. Backscattered spam from non-delivery reports can also lower sender reputation scores of the organisational mail servers potentially leading to email delivery problems.

6.1.2 Other Controls Which Can be Implemented to Limit Information Disclosure

There are various additional controls which can be implemented to limit information disclosure, such as changing default SMTP banners on filtering products and stripping information such as received headers from outbound email.

It is recommended that inbound messages are not modified to try to remove threats, as this can result in additional disclosure when messages are bounced, or failure to correctly remove all threats. Messages containing threats should be quarantined or deleted, with a separate message to the intended recipient (if required), though the original sender address should not be spoofed.

Implementing stripping of received headers on outbound messages (where this is supported by the product or service) is best done at the boundary, the last email server before the message is delivered. To prevent inbound headers from being revealed, it is also best to ensure that no auto-responders contain the original message or header attached (as described above).

While it is generally accepted that administrative UIs should not be exposed to the Internet, email security appliances often have user portals, which are intended to be exposed to a degree, as they allow users to login to manage their spam and other quarantined messages. As these interfaces often disclose product and version information, best practice suggests these portals should not be exposed to the Internet and should only be accessible via a secure VPN.

6.1.3 Tackling Client-Side Attacks

Encrypted attached documents and archives can present a significant risk to organisations in terms of threats in inbound email. As filtering solutions typically cannot be configured to decrypt these attachments, often policies are configured to let these attachments pass unprocessed. This introduces a loophole in email filtering which attackers can easily exploit. Uncontrolled use of encrypted attachments should be strictly limited by policy where possible.

In addition to filtering known malware and executable code, recent exploits should ideally be identified especially for documents, where proof-of-concept code is in the public domain. As an additional form of mitigation, sensitive organisations may wish to additionally filter high-risk document types such as PDFs and macro-enabled office documents. It may be difficult for an organisation to apply this policy to all internal recipients (due to the negative effect on business processes) though it may be possible to apply stricter policies to some groups of users to reduce overall risk.

7 Appendix

7.1 Sample Output From MailFEET for a Simple Message Set

This is a small sample output showing various test messages sent, and the resulting outcome for a typical email filtering policy assessment. Note that different results are discovered for different message types, but that a subset of outcomes needed some manual confirmation from the evidence files (in a few cases, analysis can be challenging to automate programmatically, especially where messages are modified when processed).

Test case	Title	Result	Evidence message filename
1	Echo	No threat	./messages/20140417110531-1.eml
2	Echo 1 TXT	No threat	./messages/20140417110539-3.eml
3	Echo 2 TXT	No threat	./messages/20140417110537-2.eml
4	Echo (from file)	No threat	./messages/20140417110542-4.eml
5	Exe	Notified recipient	./messages/20140417110548-5.eml
6	Exe renamed TXT	Notified recipient	./messages/20140417110551-6.eml
7	Exe renamed DOC	Notified recipient	./messages/20140417110600-7.eml
8	Encrypted GPG	Delivered	./messages/20140417110620-12.eml
9	Exe in DOCX	Notified recipient	./messages/20140417110619-11.eml
10	Exe in DOC	Notified recipient	./messages/20140417110635-15.eml
11	Exe in ODT	Notified recipient	./messages/20140417110635-16.eml
12	Exe in RTF	Notified recipient	./messages/20140417110616-8.eml
13	ZIP	No threat	./messages/20140417110627-13.eml
14	ZIP	No threat	./messages/20140417110617-9.eml
15	BAT Script in DOCX	Delivered	./messages/20140417110637-17.eml
16	BAT Script in DOC	Delivered	./messages/20140417110632-14.eml
17	BAT Script in ODT	Check message?	./messages/20140417110619-10.eml
18	BAT Script in RTF	Delivered	./messages/20140417110637-18.eml
19	Password XLS (old)	Delivered	./messages/20140417110638-19.eml
20	Password XLSB	Delivered	./messages/20140417115810-23.eml
21	Password XLSX	Delivered	./messages/20140417115758-20.eml
22	Password XLS	Delivered	./messages/20140417115811-24.eml
23	Password ODS	Delivered	./messages/20140417115805-22.eml
24	Password XLSM	Delivered	./messages/20140417115813-28.eml
25	Password DOCX	Delivered	./messages/20140417115805-21.eml
26	Password DOC	Delivered	./messages/20140417115812-27.eml
27	Password DOCM	Delivered	./messages/20140417115812-26.eml
28	Password ODT	Delivered	./messages/20140417115811-25.eml
29	Password PPSM	Delivered	./messages/20140417115814-29.eml
30	Password PPTX	Delivered	./messages/20140417115815-30.eml
31	Password PPSX	Delivered	./messages/20140417115820-34.eml
32	Password PPTM	Delivered	./messages/20140417115816-31.eml

33	Password ODP	Delivered	./messages/20140417115818-32.eml
34	Password PPT	Delivered	./messages/20140417115820-35.eml
35	Link DOC by IP	Check message?	./messages/20140417115822-38.eml
36	Link DOC by hostname	Check message?	./messages/20140417115818-33.eml
37	Link EXE by IP	Check message?	./messages/20140417115820-36.eml
38	Link EXE by hostname	Check message?	./messages/20140417115821-37.eml
39	Link DOC by IP	Check message?	./messages/20140417115823-39.eml
40	Link DOC by hostname	Check message?	./messages/20140417115827-42.eml
41	Link EXE by IP	Check message?	./messages/20140417115826-40.eml
42	Link EXE by hostname	Check message?	./messages/20140417115826-41.eml
43	Macro XLSM	Delivered	./messages/20140417115830-43.eml
44	Macro XLS	Delivered	./messages/20140417115847-48.eml
45	Macro XLS (old)	Delivered	./messages/20140417115837-44.eml
46	Macro XLSM renamed XLS	Delivered	./messages/20140417115847-49.eml
47	VBS Script in DOC	Delivered	./messages/20140417115847-50.eml
48	VBS Script in DOCX	Delivered	./messages/20140417115837-45.eml
49	VBS Script in ODT	Notified recipient	./messages/20140417115838-46.eml
50	VBS Script in RTF	Modified?	./messages/20140417115854-54.eml
51	Truncated ZIP1	Delivered	./messages/20140417115842-47.eml
52	Truncated ZIP2	Delivered	./messages/20140417115848-51.eml
53	Password ZIP1	Delivered	./messages/20140417115848-52.eml
54	Password ZIP1	Delivered	./messages/20140417115849-53.eml
55	EXE and truncated ZIP	Filtered	No Reply
56	EXE and password ZIP	Delivered	./messages/20140417115908-55.eml

Figure 23 Example output from a test of a single domain with a number of test messages