# USING DMARC TO IMPROVE YOUR EMAIL REPUTATION

*Terry Zink*
Microsoft, USA

Email tzink@microsoft.com

## ABSTRACT

In 2012, the world of email filtering created a new tool to combat spam and phishing: DMARC [1, 2]. DMARC, or Domain-based Message Authentication, Reporting & Conformance, is a technology that is designed to prevent spammers from forging the sender, thus making brands more resistant to abuse. However, its most powerful feature is the built-in reporting mechanism that lets brand owners know they are being spoofed.

DMARC has its upsides, and it is very useful for preventing spoofing, but it also has some drawbacks – it will flag some legitimate email as spam, and it will cause some short-term pain.

In addition, DMARC is difficult to set up for a large organization with a decentralized email infrastructure. Many divisions do not have email expertise, but they still need their email delivered.

This article discusses the advantages and drawbacks of DMARC. It also discusses the process that *Microsoft* went through to catalogue all of its domains in order to ensure that all of them could pass basic authentication checks. This involved creating DMARC records, sorting through legitimate and malicious sources of spoofed email, and working with teams to ensure that they could authenticate in the future.

## 1. BACKGROUND

People who are new to email security are often surprised to see just how insecure email actually is. When shown how easy it is to spoof a message, they are taken aback. Most people assume that you have to login and enter your username and password in order to send a message. This is not so; sending an email is as simple as connecting to a mail server using the SMTP protocol and transmitting a message. Furthermore, anyone can put anything as the From: address of the message. There is nothing to prevent a sender from doing this.

Figures 1 and 2 are both messages I received 'from' *PayPal* during the past three months. Which one is real and which one was faked?

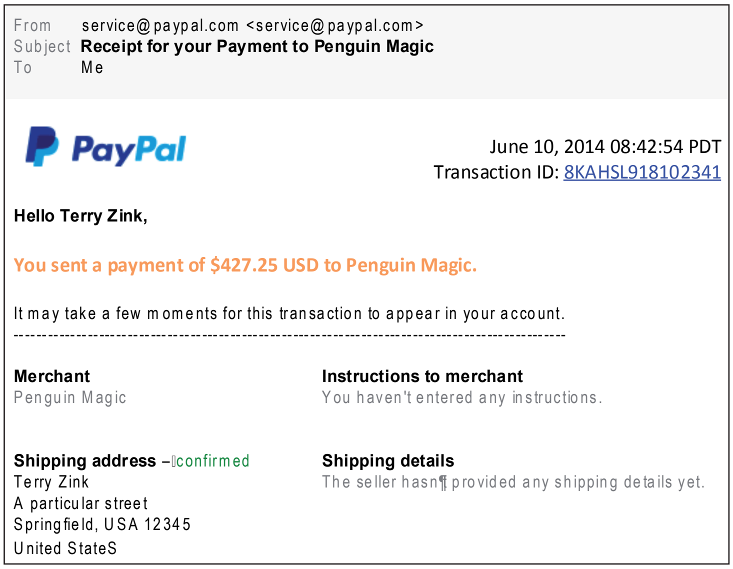
As an end-user, it is great that I can use online payment systems, but when spammers figured out that they could put a trusted brand into the From: address of an email, they discovered that users would believe that the email really came from that organization. Over the past decade, the email security industry has come up with ways to mitigate this problem using two primary technologies.

### 1.1 Terminology

In email, people naturally thing that the sender of the message is the one in the From: field – the one that they see in their email client. For example, suppose that you are a travel enthusiast and you receive the email shown in Figure 3. You received the message 'from' oceanic@news.oceanicairlines.com, right? Wrong.

In email, there are two 'From' addresses:

1. The SMTP MAIL FROM, otherwise known as the RFC 5321.MailFrom [3]. This is the email address to which the



From service@paypal.com <service@paypal.com>
Subject **Receipt for your Payment to Penguin Magic**
To Me

June 10, 2014 08:42:54 PDT
Transaction ID: 8KAHSL918102341

**Hello Terry Zink,**

**You sent a payment of $427.25 USD to Penguin Magic.**

It may take a few moments for this transaction to appear in your account.

-----------------------------------------------------------------------------------------------

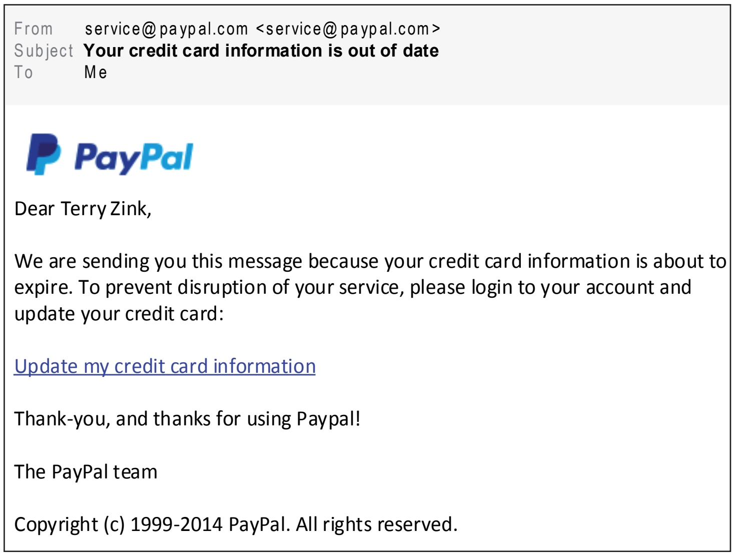
**Merchant**
Penguin Magic

**Instructions to merchant**
You haven't entered any instructions.

**Shipping address** – confirmed
Terry Zink
A particular street
Springfield, USA 12345
United StateS

**Shipping details**
The seller hasn¶ provided any shipping details yet.

*Figure 1: Is this a real message from PayPal about my recent purchase of magic supplies?*



From service@paypal.com <service@paypal.com>
Subject **Your credit card information is out of date**
To Me

Dear Terry Zink,

We are sending you this message because your credit card information is about to expire. To prevent disruption of your service, please login to your account and update your credit card:

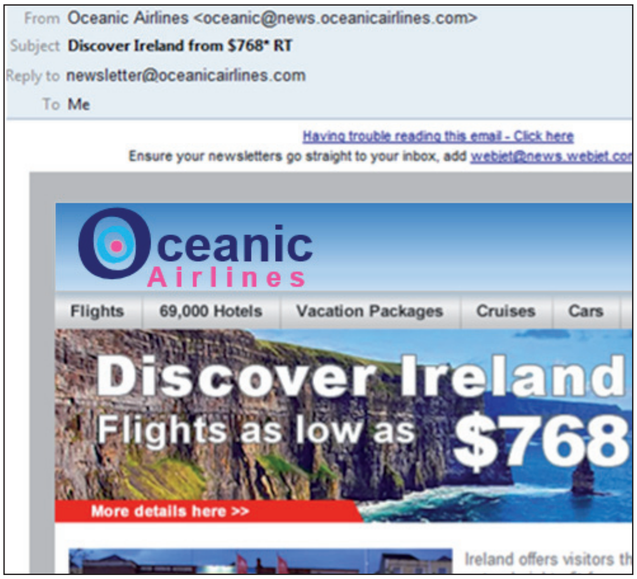
Update my credit card information

Thank-you, and thanks for using Paypal!

The PayPal team

Copyright (c) 1999-2014 PayPal. All rights reserved.

*Figure 2: Is this a real message from PayPal letting me know that my credit card will expire soon?*



From Oceanic Airlines <oceanic@news.oceanicairlines.com>
Subject Discover Ireland from $768* RT
Reply to newsletter@oceanicairlines.com
To Me

Having trouble reading this email - Click here
Ensure your newsletters go straight to your inbox, add webiet@news.webiet.cor

*Figure 3: An example email with two different ‘From’ addresses.*

bounced message will be delivered if the message cannot be delivered. It is this email address that goes into the Return-Path in the message headers.

2. The From: address in the message headers, otherwise known as the RFC 5322.From. This is the email address that is displayed in the email client.

Much of the time, the 5321.MailFrom and 5322.From addresses are the same. This is typical for person-to-person communication and is what people usually want to add safe senders for. However, when email is sent on behalf of someone else, the two ‘From’ addresses are frequently different. This happens most often for bulk email.

In the email example shown in Figure 3, the sender (From: address) is oceanic@news.oceanicairlines.com. However, Oceanic Airlines has contracted Big Communications, Inc. to send out bulk email on its behalf. The 5321.MailFrom is campaign-0152121-oceanic.airlines@bigcommunications.com, and that is the address to which email bounces are delivered for email campaign tracking. However, to the end-user, the message appears to have come ‘from’ Oceanic Airlines, because that’s what they see in their email client.

This difference between the 5321.MailFrom and 5322.From addresses is important!

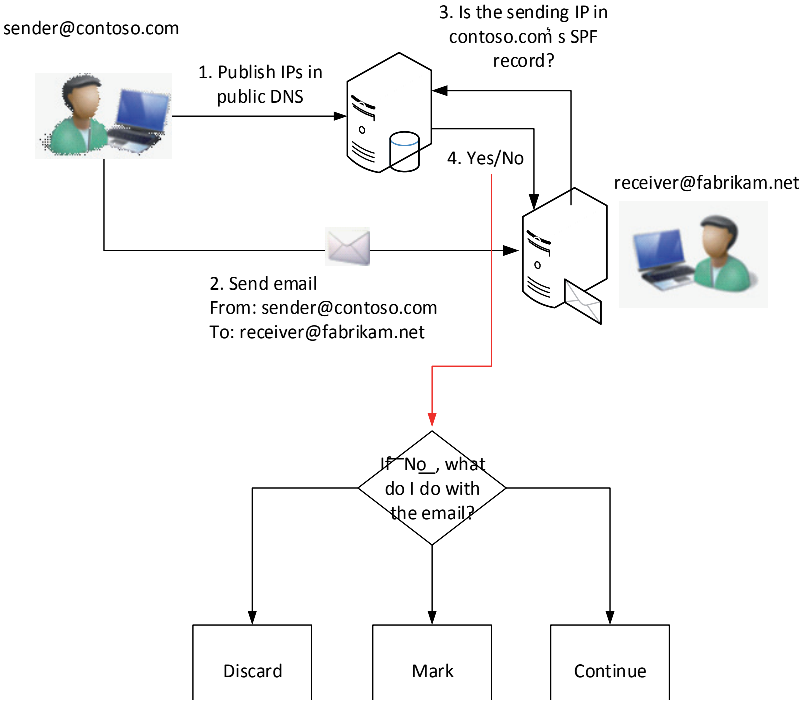
## 1.2 Sender Policy Framework – SPF

SPF is a technology that identifies the path that a message took to get to you, and whether or not that path was authorized.

Domains publish a set of IPs in their SPF records. If email comes from a domain in the 5321.MailFrom, an email receiver looks up the SPF record for that domain. It then asks the question ‘Does the IP from which this email came match any of the IPs in the SPF record?’ If it does, the message passes SPF. If it does not, SPF allows the domain owner to specify what to do with the message: hard fail, -all (give it a heavy weight in the spam filter); soft fail, ~all (give it a light weight in the spam filter); or neutral, ?all (treat it as if it had no SPF record).

Suppose we had the following SPF record:

```
contoso.com.   IN   TXT   “v=spf1 ip4:1.2.3.4 -all”
```



sender@contoso.com
1. Publish IPs in public DNS
3. Is the sending IP in contoso.com’s SPF record?
4. Yes/No
receiver@fabrikam.net
2. Send email
From: sender@contoso.com
To: receiver@fabrikam.net
If “No”, what do I do with the email?
Discard
Mark
Continue

*Figure 4: How SPF works.*

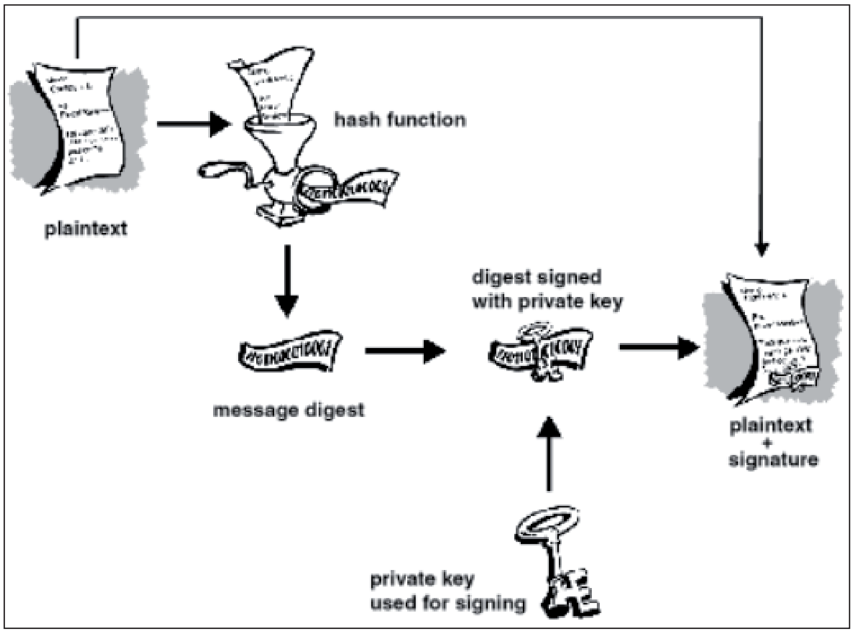


*Figure 5: How DKIM works.*

If fabrikam.net ever gets an email from contoso.com from the IP 1.2.3.4, it passes SPF and authenticates.

However, if fabrikam.net receives an email from contoso.com (in the 5321.MailFrom) from the IP 5.6.7.8, it can see that the IP is not included in contoso.com's SPF record. Fabrikam sees that the SPF policy is 'hard fail', so it can give it a heavy weight and mark it as spam for the recipient.

### 1.3 DomainKeys Identified Mail – DKIM

DKIM does not rely upon the path of the message and instead relies upon properties of the message itself. DKIM requires the generation of two encryption keys: a public key which is published in a domain's DNS record, and a corresponding private key. It creates a digital signature which is derived from the message contents and encrypted with the private key using public key encryption. It then transmits the digital signature along with the email message. The message signer tells the receiver where to look up the public key by stamping it into the d= field in the DKIM-Signature header in the message.

The receiver gets the message and extracts the digital signature. He then checks the sender's domain in the d= field in the DKIM-Signature header and looks up the public key for that domain in DNS, performing a DNS lookup. Finally, he validates the signature using the public key. If it checks out, he knows that the message really came from the sender.

Comparing DKIM with SPF, DKIM offers the following advantages:

1. **Security**: A spammer cannot spoof the message because he does not have access to the private key with which the sender signed the message. Without the private key, the public key is useless.
2. **Does not break email forwarding**: Because DKIM does not rely on the sending IP and only on the email content, email can be forwarded without affecting the DKIM validation as long as the message is not modified.
3. **Anti-tampering**: The message is not modified in transit. If even one bit within the message is changed after the sender sent it, this would change the digital signature and the message would not be able to be validated.

## 2. WHAT IS DMARC?

To understand what DMARC is used for, we first need to understand the limitations of SPF and DKIM[1].

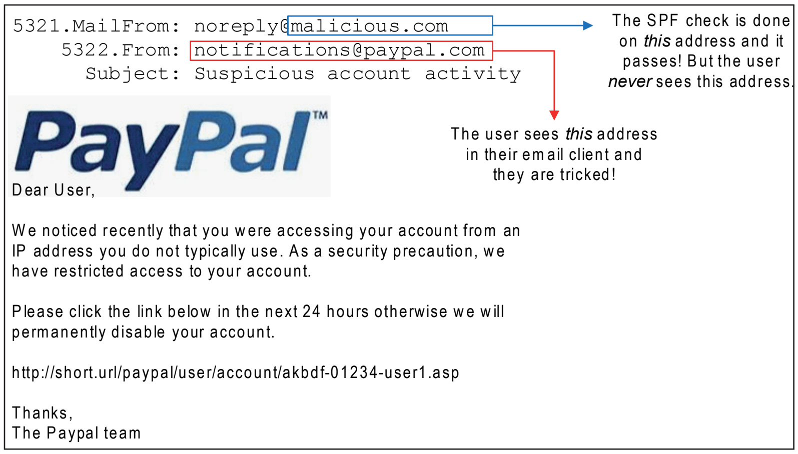
### 2.1 Weaknesses of SPF and DKIM

While SPF and DKIM are both good under certain circumstances, they suffer from a serious weakness – they do not protect the email address that the user sees in their inbox. SPF authenticates the SMTP MAIL FROM, which may or may not be the same as the 5322.From address. Meanwhile, DKIM authenticates the domain in the d= field in the DKIM-Signature, and there is no requirement for it to be the same as what the user sees.

There is nothing to stop a spammer from sending email that passes SPF or DKIM, or both, while specifying a different 5322.From address. From an anti-spam perspective, such a message will pass filtering.

This is where DMARC (or Domain-based Message Authentication, Reporting & Conformance) comes in. In the

[1] There are more drawbacks to SPF and DKIM but they are omitted from this paper.



```
5321.MailFrom: noreply@malicious.com
     5322.From: notifications@paypal.com
       Subject: Suspicious account activity
```

The SPF check is done on *this* address and it passes! But the user *never* sees this address.

The user sees *this* address in their email client and they are tricked!

Dear User,

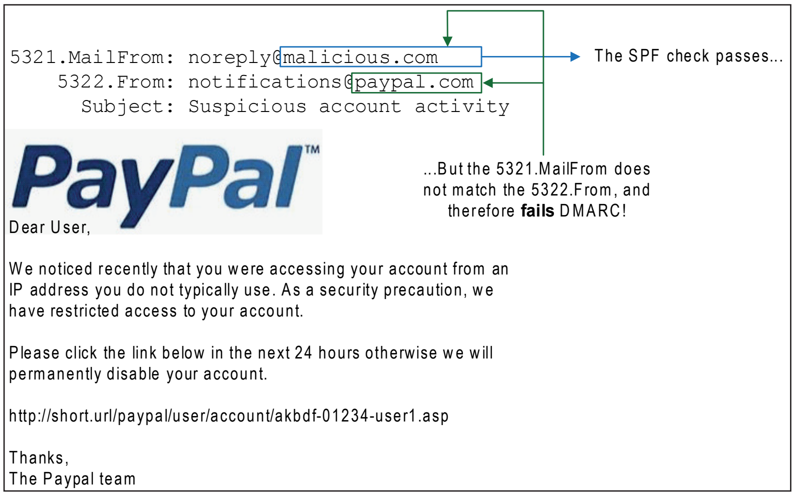
We noticed recently that you were accessing your account from an IP address you do not typically use. As a security precaution, we have restricted access to your account.

Please click the link below in the next 24 hours otherwise we will permanently disable your account.

http://short.url/paypal/user/account/akbdf-01234-user1.asp

Thanks,
The Paypal team

*Figure 6: How spammers can pass SPF but still phish users.*



```
5321.MailFrom: noreply@malicious.com
     5322.From: notifications@paypal.com
       Subject: Suspicious account activity
```

The SPF check passes...

...But the 5321.MailFrom does not match the 5322.From, and therefore **fails** DMARC!

Dear User,

We noticed recently that you were accessing your account from an IP address you do not typically use. As a security precaution, we have restricted access to your account.

Please click the link below in the next 24 hours otherwise we will permanently disable your account.

http://short.url/paypal/user/account/akbdf-01234-user1.asp

Thanks,
The Paypal team

*Figure 7: How DMARC stops phishing.*

example shown in Figure 6, the user sees the address notifications@paypal.com in their inbox and may be deceived into thinking that the message really was from *PayPal*. Had the 5321.MailFrom and 5322.From been the same email address, SPF would have detected this and the message would have been marked as spam. However, since the two addresses are different, and since email clients do not tell the end-user they are different, they can be tricked. Furthermore, DKIM only identifies a real owner of a message. If a message is unsigned, or fails to verify, receivers should treat the message as if it is unsigned. DKIM establishes identity, it does not detect phishing.

In contrast, DMARC is a technology that is designed to combat spoofing of the 5322.From address. DMARC asserts that:

1. The message must pass SPF or DKIM.
2. The 5322.From address – the one that the user sees – and the domain that is authenticated (using either SPF or DKIM) must be the same[2].

Therefore, even if a spammer tries to ‘hide’ the result of an SPF check, the fact that the authenticated domain does not match the one that the user sees will cause it to be flagged as spam.

Thus, by publishing a DMARC record, a domain owner can indicate to receivers how they should treat email ‘from’ their domain that does not authenticate.

A domain publishes a DMARC record at _dmarc.<domain>. For example, _dmarc.contoso.com. Upon receipt of a message, an email receiver checks the DMARC record of the domain in the 5322.From address. There are four options, as shown in Table 1.

DMARC is useful up to this point because it allows domains to specify what to do with spoofed messages. However, SPF does the same thing with its -all, ~all and ?all mechanism.

But DMARC has a powerful feature that the other technologies don’t – reporting back ln failures. Whenever a message fails DMARC validation, a report is sent back to the spoofed domain, indicating that a message containing that domain failed DMARC validation. This feedback report is either done on an individual basis or sent in aggregate. That is, either a copy of the failed

[2] Technically, these two domains must align. DMARC lets the domain require an exact match or a relaxed match where the organizational domains must match.

| Case | DMARC record | Requested action for receivers |
|---|---|---|
| 1 | The DMARC record doesn't exist | Continue normal filtering |
| 2 | The DMARC record indicates p=none | Treat the message as if it didn't fail DMARC |
| 3 | The DMARC record indicates p=quarantine | Mark the message as spam |
| 4 | The DMARC record indicates p=reject | Reject the message without accepting it, the end-user receives no copy of the message |

*Table 1: Four DMARC options.*

message is sent back to the forensic reporting email address, or a rolled up aggregate report containing basic information is sent back to the aggregate reporting email address.

This means that a domain can publish a DMARC record and receive all sorts of feedback. The domain owners can publish a DMARC record of p=none and then collect reports to see what would happen if they published p=quarantine or p=reject.

SPF was supposed to be a strict mechanism, but because there are so many legitimate cases of SPF failing, domains are reticent to publish strict policies. However, with DMARC, domains can take a look to see the potential consequences *ahead of* publishing a strict DMARC policy. With SPF and even DKIM, domains were unclear as to how receivers would treat their email.

In contrast, DMARC is like turning on the lights in a dark room.

## 3. BENEFITS OF DMARC

There are some obvious and less obvious benefits to implementing DMARC.

### 3.1 Decreasing phishing

#### 3.1.1 Decreasing regular phishing

The most important benefit of DMARC is its ability to decrease phishing. Users rely on the 5322.From address to see who a message is from, so if spoofed 5322.From addresses fail to make it into their inboxes, users will fall victim to these scams less often. It is important to banks that their customers do not fall victim to phishing scams, because it costs them time and money to recover lost funds.

#### 3.1.2 Decreasing spear phishing

DMARC is also important for large organizations that are targeted by spear phishing. One of the most common tactics of hackers trying to infiltrate an organization to plant an advanced persistent threat is to use phishing. They will send an 'important' message spoofing the organization itself. For example, suppose the company Contoso is an important energy company, and employee John Smith receives the following email:

```
5321.MailFrom: hacker@freewebmail.com
DKIM-Signature: d=freewebmail.com
From: important_person@contoso.com
To: john.smith@contoso.com
Subject: Could you open up and fill out this work
order?
Attachment(s): work_order_68241925.docx
```

The parts in italics are not shown to the end-user, so they can't see that the message is spoofed. It is deceptive because it appears to come from the same organization as the victim, and most people communicate regularly with others inside their company. However, if John Smith opens the message, it could trigger a zero-day exploit and grant the attacker remote access to his machine.

If Contoso had a DMARC record of p=quarantine or p=reject, on the other hand, the message would be marked as spam or rejected, and John Smith would never see it.

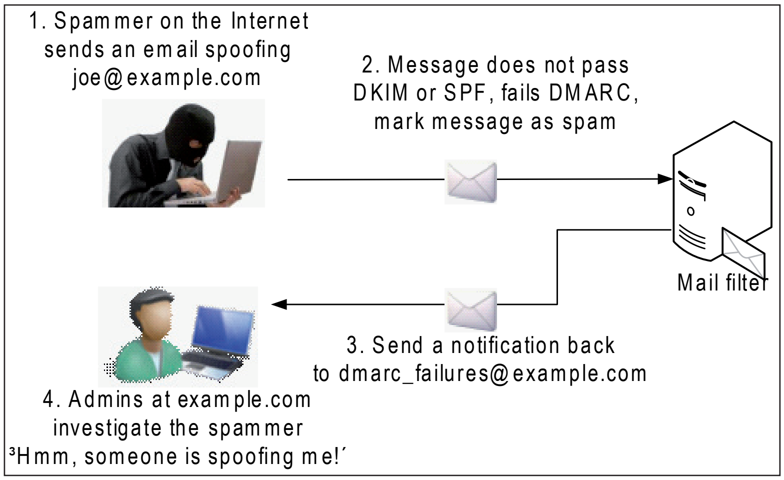

1. Spammer on the Internet sends an email spoofing joe@example.com
2. Message does not pass DKIM or SPF, fails DMARC, mark message as spam
Mail filter
3. Send a notification back to dmarc_failures@example.com
4. Admins at example.com investigate the spammer ³Hmm, someone is spoofing me!´

*Figure 8: DMARC with feedback reports.*

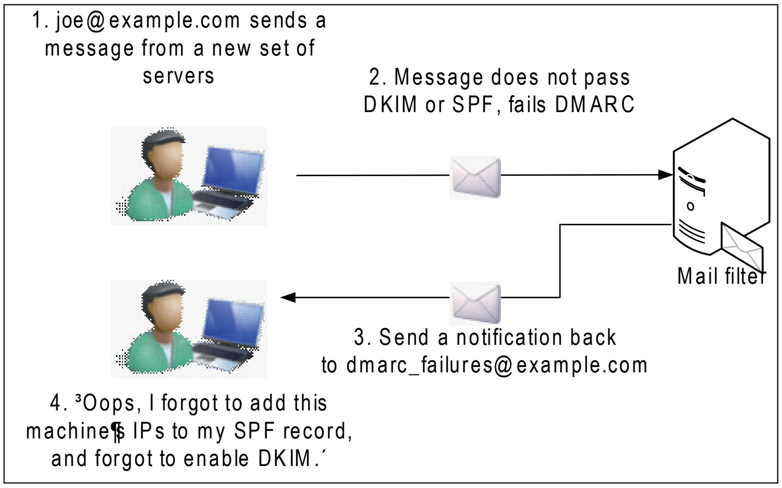

1. joe@example.com sends a message from a new set of servers
2. Message does not pass DKIM or SPF, fails DMARC
Mail filter
3. Send a notification back to dmarc_failures@example.com
4. ³Oops, I forgot to add this machine¶s IPs to my SPF record, and forgot to enable DKIM.´

*Figure 9: Using DMARC to detect a misconfiguration.*

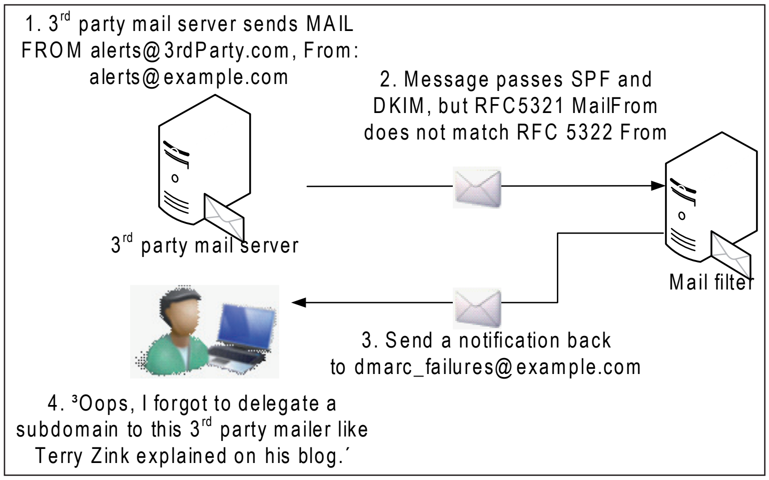

1. 3rd party mail server sends MAIL FROM alerts@3rdParty.com, From: alerts@example.com
3rd party mail server
2. Message passes SPF and DKIM, but RFC5321 MailFrom does not match RFC 5322 From
Mail filter
3. Send a notification back to dmarc_failures@example.com
4. ³Oops, I forgot to delegate a subdomain to this 3rd party mailer like Terry Zink explained on his blog.´

*Figure 10: Using DMARC to inventory all third-party emailers [4].*

This is how DMARC helps to reduce spear phishing attacks. While there is no technology that will completely eliminate either regular phishing or spear phishing, DMARC is an important layer that minimizes one of the ways in which hackers gain access to organizations.

## 3.2 Detect misconfigurations

If a new server that sends outbound email is brought online, and that server doesn’t sign with DKIM, or the IP addresses have not been added to the SPF record, DMARC can proactively notify the sender that authentication is failing (see Figure 9).

## 3.3 Inventory third-party mailers

In order to get SPF records under control, DMARC can be used to inventory all the IPs that are sending email ‘as’ a brand. With this information, organizations can add them to their SPF records and eventually publish -all in their SPF record since they will have full confidence in who sends email legitimately ‘as’ them.

### *Before inventorying third-party emailers*

SPF record: example.com IN TXT “v=spf1 1.2.3.0/24 ~all

In this example, the soft fail is published because example.com doesn’t want to lose any legitimate email from third parties that send email as them, but which don’t authenticate.

### *After inventorying third-party emailers*

SPF record: example.com IN TXT “v=spf1 1.2.3.0/24 -all”

SPF record: 3rdparty.example.com IN TXT “v=spf1 include:3rdparty.com –all”

The SPF record now has a hard fail, which means that email receivers that do not support DMARC but which do support

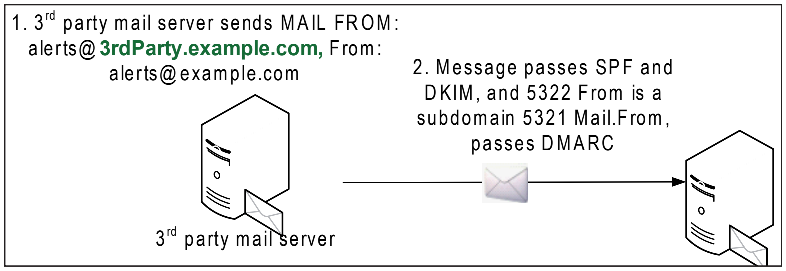

1. 3rd party mail server sends MAIL FROM: alerts@3rdParty.example.com, From: alerts@example.com
2. Message passes SPF and DKIM, and 5322 From is a subdomain 5321 Mail.From, passes DMARC
3rd party mail server

*Figure 11: After inventorying third-party emailers.*

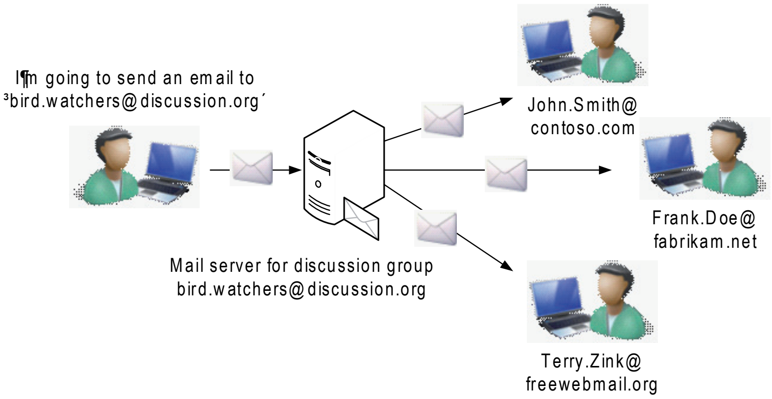

I¶m going to send an email to ³bird.watchers@discussion.org´
Mail server for discussion group bird.watchers@discussion.org
John.Smith@ contoso.com
Frank.Doe@ fabrikam.net
Terry.Zink@ freewebmail.org

*Figure 12: A mailing list.*

SPF can assign a heavier weight. This helps prevent a brand (e.g. *Microsoft*) from being spoofed by spammers.

## 4. DRAWBACKS OF DMARC

DMARC is a major step forward. However, it has some limitations.

### 4.1 Homoglyph attacks

DMARC is designed to catch the case where, for example, a spammer spoofs '@paypal.com'. However, it does not catch the case where the spammer changes the domain to make it *look like* the target domain (e.g. '@paypa1.com'), or where the target domain occurs somewhere within the email address (e.g. '@paypal.com.spam.net').

Organizations can prevent the first type of spoofing by purchasing domains that sound or look very similar to their own, and then publishing SPF and DMARC records for those domains, indicating that they send no email. However, they cannot register all possible lookalikes, nor can they prevent the case where the phisher uses the target as a subdomain.

DMARC does not address this issue, but at the same time, these attacks are not the most common.

### 4.2 Mailing lists

In cases where a message is 'spoofed' for legitimate reasons, it will fail DMARC. This occurs most frequently in mailing lists or distribution groups. When you send a message to the list, the message goes to the group alias, which then relays the message to all recipients on the list. The message appear to have come 'from' the original sender, but in reality it has come from the distribution list's mail server. The original sender appears in the 5322.From address as a convention.

In the diagram shown in Figure 12, when tom.user@example.com sends a message to the discussion group, this is what is happening:

```
5321.MailFrom: tom.user@example.com
5322.From: tom.user@example.com
To: bird.watchers@discussion.org
DKIM-Signature: d=example.com
_dmarc.example.com = p=reject
Subject: Has anyone seen this new bluejay lately?
```

The mail server at discussion.org correctly sees that this message passes DMARC because it passes SPF and DKIM, and the domain in the 5322.From address aligns with both the SPF-validated domain and the DKIM-validated domain.

However, when the message is relayed from the discussion list's mail server, it looks like this:

```
5321.MailFrom: bird.watchers@discussion.org
5322.From: tom.user@example.com
To: <individual end user>
DKIM-Signature: d=discussion.org
_dmarc.example.com = p=reject
Subject: [bird watchers] Has anyone seen this new
bluejay lately?
```

In the case of this relayed message, the message passed SPF (discussion.org) and DKIM (d=discussion.org) checks. However, the domain in the 5322.From field (example.com) aligns with neither of those domains. Because example.com publishes p=reject, the email will be marked as spam or rejected. This is despite the fact that the message is entirely legitimate.

Even if the original message was DKIM-signed, the content of the message has been modified – the subject line has been changed to insert a tag to make it useful to all the other users on the list. Mailing lists have been doing this for decades – they modify the contents of a message to make it more useful to its recipients. Unfortunately, doing this invalidates the original DKIM signature, which means that anyone with a domain that publishes p=reject cannot send email to mailing lists if there are other recipients on the list that perform DMARC verification (i.e. just about all of them).

If a domain that publishes a DMARC record of p=reject joins a mailing list, the following is what normally occurs:

1. The user who sends to the list and whose message is relayed to the rest of the list's recipients will see their message rejected at the recipients' mail servers.
2. The mailing list will see many different rejections from these recipients for this user. Many mailing lists will then automatically unsubscribe the user from the list, erroneously believing that email address is no longer valid.

Neither of these two behaviours is desirable, and this loss of legitimate email is a major drawback for DMARC.

| **Option 1 –** Do nothing and let domains that publish p=reject live with the consequences | |
|---|---|
| **Advantages** | **Drawbacks** |
| Requires no code changes. | Doesn't address the problem.<br>Major brands already publish p=reject and it would be a net benefit for users of major brands to be able to join discussion lists. |

| **Option 2 –** Don't permit domains with p=reject onto mailing lists | |
|---|---|
| **Advantages** | **Drawbacks** |
| Prevents negative consequences of allowing domains with p=reject onto lists. | Same as above. |

| **Option 3 –** Don't modify messages when sending to mailing lists | |
|---|---|
| **Advantages** | **Drawbacks** |
| Messages that are signed with DKIM can participate in mailing lists. | Loses useful features of mailing lists: message modification is important. |

| **Option 4 –** Extend DMARC protocol so that it supports mailing lists | |
|---|---|
| **Advantages** | **Drawbacks** |
| Potentially scales to all DMARC broken cases. | Cost/benefit ratio unclear of extending DMARC. |

| **Option 5 –** Mailing lists should reformat the message to prevent DMARC failures | |
|---|---|
| **Advantages** | **Drawbacks** |
| Allows users to join mailing lists. | Can confuse end-users.<br>Turns mailing list server into a mail relay which may not be desired behaviour. |

| **Option 6 –** Email receivers should be selective about how they enforce p=reject - send it to Junk or even skip enforcing it from known good emailing lists | |
|---|---|
| **Advantages** | **Drawbacks** |
| Doesn't bounce domains with p=reject on mailing lists, avoids most negative consequences. | Defeats the purpose of rejecting email, user can still access phishing messages. |

| **Option 7 –** Maintain a whitelist of known mailing lists | |
|---|---|
| **Advantages** | **Drawbacks** |
| Makes use of heuristics that most email receivers do anyhow. | Expensive to maintain. |

*Table 2: The numerous options to work around DMARC breaking lists.*

### 4.3 Workarounds

The problem of mailing lists is one of the reasons why previous efforts to tie the 5322.From address to an authenticated domain failed. Until April 2014, the conventional wisdom was 'Domains that publish p=reject shouldn't participate in mailing lists. Use a non-highly targeted domain instead.' This held true, for the most part, until both *Yahoo* and *AOL* started publishing p=reject.

So, how can this be fixed? There are numerous options[3], which are shown in Table 2.

None of these solutions are easy to implement, and none of them are ideal. However, the best ones are the last two. They have the drawback of forcing implementations to stop treating DMARC as a yes/no response to the question 'Is this message spoofed and if so, what should I do with it?' Instead, the answer becomes 'It depends'.

Since June 2014, the DMARC Working Group has been hard at work, trying to come up with ways to avoid interfering with legitimate messages. There is currently no consensus, and until there is, DMARC will continue to cause problems for certain cases of legitimate email.

## 5. CASE STUDY: MICROSOFT

### 5.1 The problem

*Microsoft* is a large organization, within which there are many different business units that send email to its customers. Some of them include:

- Bing Rewards
- HealthVault
- Microsoft Volume Licensing
- MS Blog Admins
- Photosynth
- Visual Studio
- Windows Phone
- Xbox
- Xbox Live Enforcers

The sending of email to the company's customers is not universally coordinated across business units within the organization. Some teams use third-party bulk email providers to send email. Some use another internal team dedicated to sending bulk email. Others have set up their own mail servers and call APIs that are native to Powershell. And still others use third-party hardware-hosting providers that also provide email-sending services.

Thus, while microsoft.com is prone to being spoofed, the problem is that nobody wants to see their legitimate email marked as spam by publishing aggressive DNS records indicating what to do with spoofed email. That is, HealthVault needs to have its email delivered to end-users, so does Xbox Live Enforcers, so does Visual Studio, and so on. For that reason, *Microsoft* publishes the following SPF record:

```
microsoft.com. IN   TXT   "v=spf1 <list of all records
~all"
```

*Microsoft* publishes a soft fail in its SPF record, which means 'Accept the email but mark it in some way'. In reality, most email receivers treat a soft fail as a very light weight in their spam-filtering algorithms. Thus, when an email server passes SPF, it is a way to identify a legitimate sender. However, a soft fail is a very weak indicator when detecting malicious spoofing. It provides very little anti-phishing protection.

In December 2013, *Microsoft* embarked on a program to publish a hard fail in its SPF. This would require putting together an inventory of *all* internal teams that were sending out emails that were failing SPF checks, and getting them to send their emails in a manner that passed SPF so that we could be more aggressive with malicious email without interfering with legitimate, but unauthenticated, email.

This was no small feat, and it would not have been possible without DMARC. While DMARC is an anti-phishing technology that can reject or mark as spam any message that fails to authenticate, its sending of reports to the spoofed domain is vital since it can be deployed in passive mode by publishing a policy of p=none, enabling a domain to receive DMARC reports. They can then use these reports to track down every single point-of-origin both within and outside their infrastructure to pinpoint which teams are not authenticating properly.

That is exactly what *Microsoft* did.

### 5.2 Working towards authentication

#### *Step 1 – Decide how to receive DMARC reports*

Because *Microsoft* is such a large company, the number of reports that would be received would be very large. There may be hundreds, thousands, or even hundreds of thousands of DMARC reports per day. Luckily, there are companies that specialize in the collection of DMARC reports and in presenting them to domain owners in a way that is easily consumable – they can be searched, sorted, and analysed.

*Microsoft* enlisted a third-party provider to assist with this[4].

#### *Step 2 – Publish a DMARC record*

The next step is to publish a DMARC record in DNS. *Microsoft* published the following:

```
_dmarc.microsoft.com.    IN        TXT     "v=DMARC1;
p=none; pct=100; rua=mailto:d@...; ruf=mailto:d@...;
fo=1"
```

This DMARC record says:

1. Do not take action if a message fails DMARC.
2. Send an aggregate report and a forensic report for *any* alignment failure – whether the message fails SPF, DMARC, or organizational alignment.

[3] A full list of workarounds is available at: http://wiki.asrg.sp.am/wiki/Mitigating_DMARC_damage_to_third_party_mail.

[4] This is not an official endorsement of this particular third company. If you want to know which one was used, please see our DNS record.

Once that record was published, all third parties that support DMARC would start sending reports back to the ruf and rua addresses.

### Step 3 – Sort through the DMARC reports for IPs that are used for corporate traffic

1. Once DMARC reports started pouring in, we had to step through them and sort through which ones were bad and which ones were misconfigurations.

   *Microsoft* sends its outbound corporate traffic through Exchange Online Protection (EOP). We noticed that there were a lot of SPF failures attributed to mail coming through EOP's outbound IPs. It turned out that *Microsoft* had so many domains listed in its SPF record, that it resulted in the maximum number of DNS lookups prescribed by SPF (10) being exceeded. We cut out some of the extraneous records and waited for more reports.

2. Once the SPF/DNS lookup limits were resolved, we discovered that there were a lot of failures on NDR messages, that is, messages with a 5321.MailFrom of <>. We realized that we needed to create SPF records for the domains in the HELO/EHLO strings that the *Exchange* servers use to send email[5].

### Step 4 – Sort through the DMARC reports for IPs that are internal to the company, but failing authentication

Next, we had to find IPs that were registered to *Microsoft* as a company. Some teams had set up mail servers to send email directly from a *Microsoft*-owned IP address. However, these IPs were not in *Microsoft*'s SPF record.

To fix this, for every IP that showed up as failing SPF, we looked at the alias (e.g. vst@microsoft.com[6]). We then tracked down the team that was using the address to send email and asked them either to move to the internally supported email-sending platform, or if that was not possible, we would add the IP range to *Microsoft*'s SPF record.

This process was repeated for several teams.

### Step 5 – Sort through the DMARC reports for IPs that are external to the company and failing authentication

Some teams at *Microsoft* were using third-party hardware to send automated email. Most of this email was transactional in nature.

We decided that *Microsoft* would not put third-party IPs into the SPF record for @microsoft.com. Instead, we would delegate a subdomain, e.g. email.microsoft.com. All third-party IPs would go into that subdomain.

We then tracked down every team that used third-party hardware to send email and asked them either to move their mailing software to use the internally supported email platform, or move their email address from <localpart>@microsoft.com to <localpart>@email.microsoft.com. We then added their IPs to the SPF record for that domain.

### Step 6 – Update DKIM keys

*Microsoft* does use some third-party email service providers to send bulk email. However, when researching the teams, we found that some of them were sending DKIM-signed email with keys that had been generated several years ago.

We decided to update them. We contacted the third-party provider and got them to generate a new public/private DKIM key pair. We then published the public key in DNS at a new domain and got them to test with the new private key. At an agreed date, we switched over to the new key and retired the old one.

### Step 7 – Update the SPF record to a hard fail

The next step is to update *Microsoft*'s SPF record to a hard fail. Because of the potential conflict with sales staff and a busy end of the fiscal year, *Microsoft* decided to put the change to a hard fail on hold until after the fiscal year, to mitigate risk to the service.

However, when all was said and done, after an eight-month process, *Microsoft* updated its SPF record from a soft fail to a hard fail. This would not stop all spoofing of *Microsoft*'s domain, but it would certainly stop some.

### Step 8 – Publish a DMARC record of p=quarantine

Publishing a stricter DMARC record is on the horizon. But first, *Microsoft* needs to sign all of its corporate traffic with DKIM. That is still a work in progress and has not yet been accomplished. However, it is planned for a future release.

Without DMARC's reporting feedback, this process would not have been possible.

## CONCLUSION

DMARC is a very useful technology for preventing spoofing; it is the biggest step forward we have seen in five years for combating phishing. Domains that have implemented it with a strict rejection policy report that phishing of their domains has declined significantly since they published the aggressive policy. DMARC's strength is that it helps domains preview what will happen when they do lock down their domain, so as to reduce false positives before they occur.

Yet DMARC still has some issues to work out. It flags legitimate email as spam and there are no elegant workarounds yet. However, as these issues get sorted out, the end result will be a more secure environment for email.

## REFERENCES

[1] DMARC information page. http://dmarc.org/.

[2] DMARC official requirements. https://datatracker.ietf.org/doc/draft-kucherawy-dmarc-base/.

[5] If the 5321.MailFrom is empty, SPF uses the domain in the HELO/EHLO string.

[6] Fictitious email address.

[3] A Reputation Response Set for Email Identifiers. http://tools.ietf.org/html/rfc7073.

[4] See my blog series on DMARC and third-party emailers: http://blogs.msdn.com/b/tzink/archive/2013/04/27/how-to-setup-dmarc-records-if-you-are-outsourcing-some-or-all-of-your-email-part-1.aspx and http://blogs.msdn.com/b/tzink/archive/2013/04/27/how-to-setup-your-dmarc-records-if-you-are-outsourcing-some-or-all-of-your-email-part-1.aspx.