DMARC

“How to Understand the Fine Details”

Tyler Roberts

troberts@securit360.com

SecurIT360
The physics of securing IT

SecurIT360
OFFENSIVE SECURITY

DMARC Rundown

"Why do we need DMARC?" - DMARC allows domain owners to protect their domain(s) from unauthorized use by fighting phishing, spoofing, CEO fraud, and Business Email Compromise.

"What is DMARC?" - A DMARC record is a text entry within a DNS record that tells the world your email domain's policy after checking SPF and DKIM status.

"How does DMARC work?" – DMARC encompasses both SPF and DKIM. When an email is sent the receiving sender will check to see if the DKIM signature of the incoming mail matches and that the sender is an approved sender in the SPF record. If one of, or neither of these parameters are met, then DMARC states what the recipient of the mail should do with it. The three policies are listed below:

The three DMARC policies are:

- **p=none**

  Monitors your email traffic. No further action is taken.

- **p=quarantine**

  Sends unauthorized emails to the spam folder, when quarantine is enabled as the p, messages aren't rejected, they're sent to the RUA for review.

- **p=reject**

  The final policy and the goal of implementing DMARC. This policy ensures that unauthorized email doesn't get delivered at all.

A DMARC record can also tell email servers to send XML reports back to the reporting email address listed in the DMARC record (aka the RUA as listed below). These reports provide insight on how your email is moving through the ecosystem and allow you to identify everything that is using your email domain.

"How do I implement DMARC?" – I typically like to tell people that DMARC is a slow rollout, meaning that organizations should not opt to a reject policy to start. The best place to start is with a none policy and implement an rua to get aggregate reports as an insight to how mail is flowing through the ecosystem in order to understand who is sending on your behalf. Once you know who the approved senders on your behalf are, you can update your SPF record and then move to a quarantine policy and work to a reject policy.

Some important definitions for DMARC include the following:

- **V:** The version of DMARC that is running
- **P**: Policy for organizational domain - aka what your server does with mail it does not recognize - reject - quarantine (doesn't reject or accept) or none (meaning all mail is allowed, there is no filter enabled.)
- **sp**: Subdomain policy
- **rua:** The email address where the DMARC reports are sent to, these reports provide insight into several things which include: date and time range of the report, the domain, the IP address that sent the message, Whether SPF and DKIM passed or failed, the DMARC policy that is applied.
- **ruf:** Originally created with the intent to provide domain owners with redacted copies of emails that did not pass DMARC compliance. This is NOT recommended because of privacy concerns involving partial or inadequate redaction.

Below is an example of a correctly formatted DMARC entry. Note, that the end goal of the "p" value is "*reject*".

"v=DMARC1; p=none; rua=mailto:dmarc@yourdomain.com"

SPF Rundown

"What is SPF and why do I need it in my DNS?" - Sender Policy Framework (SPF) hardens your DNS servers by restricting who can send emails from your domain - making your mail server more secure.

"How does it help?" - SPF can prevent domain spoofing. It enables your mail server to determine when a message came from the domain that it uses.

Key symbols to note include:

- **v:** SPF version. This tag is required and must be the first tag in the record.
- **ip4:** Authorize mail servers by IPv4 address or address range
- **ip6:** Authorize mail servers by IPv6 address or address range can be in two formats (ip6:3FFE:0000:0000:0001:0200:F8FF:FE75:50DF) or (ip6:2001:db8:1234::/48)
- **a:** Authorize mail servers by domain name
- **mx:** Authorize one or more mail servers by domain MX record
- **include:** Authorize third-party email senders by domain
- **all:** Specifies that all incoming messages match. Any mechanism that comes after all mechanism in an SPF record is ignored

There are two different types of "all" at the end

- **~all**: the server will accept messages from senders not in the SPF but mark them as suspicious
- -**all:** receiving servers may reject messages from senders that aren't in your SPF record

An example of a complete SPF:

v=spf1 mx ip4:209.85.220.69 ip4:35.195.220.65 ip4:68.148.157.91 ip4:104.131.221.179 ip4:18.223.170.225 include: _spf.google.com include:spf.mailjet.com ~all

DKIM Rundown

"What is DKIM and why do I need it?" - In short, DKIM stores a txt file in your DNS that contains a public key that is used by receiving mail servers to verify a messagexs signature.

"What is a DKIM signature?" - DKIM gives emails a signature header that is added to an email and secured by encryption, the signature contains all the necessary information for the email server to verify that the signature is real.

"How does DKIM work?" - When an inbound mail server receives a message, it will detect the DKIM signature and look up the sender's public DKIM key in DNS.

When an inbound mail server receives a message, it will detect the DKIM signature and look up the sender's public DKIM key in DNS. If the DKIM provided matches what's in the DNS, then your key is valid.

"Why can't I use just DKIM or just SPF?" - DKIM on its own isn't a reliable way of authenticating the identity of the email sender and does nothing to prevent the spoofing of the domain visible in the header of the email.

DMARC solves the problem by guaranteeing that the domain the end user sees is the same as the domain that is validated by DKIM and SPF.

"What does a correctly configured DKIM look like?"

dk1024-2012._domainkey.returnpath.com. 600 IN TXT "v=DKIM1\;
p=MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQC1TaNgLlSyQMNWVLNLvyY/neDgaL2oqQE8T5illK
qCgDtFHc8eHVAU+nlcaGmrKmDMw9dbgiGk1ocgZ56NR4ycfUHwQhvQPMUZw0cveel/8EAGoi/UyPmqfcP
ibytH81NFtTMAxUeM4Op8A6iHkvAMj5qLf4YRNsTkKAV;"

- **s:** Indicates the selector record name used with the domain to locate the public key in DNS
- **d:** Indicates the domain used with the selector record (s=) to locate the public key
- **p:** Indicates the public key used by a mailbox provider to match to the DKIM signature.

In the example used above:

The selector (s=): dk1024-2012

The domain (d=): returnpath.com

The version (v=): DKIM1

The public key (p=):
MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQC1TaNgLlSyQMNWVLNLvyY/neDgaL2oqQE8T5illKqC
gDtFHc8eHVAU+nlcaGmrKmDMw9dbgiGk1ocgZ56NR4ycfUHwQhvQPMUZw0cveel/8EAGoi/UyPmqfcPib
ytH81NFtTMAxUeM4Op8A6iHkvAMj5qLf4YRNsTkKAV

References

https://dmarcian.com/why-dmarc/

https://www.clusterednetworks.com/spf-dmarc-record-cheatsheet

https://dmarc.org/overview/

DMARC Checker

https://mxtoolbox.com/DMARC.aspx

Catch our weekly podcast:

https://offsec.blog/category/podcast/