DKIM Overview

DKIM – which stands for Domain Keys Identified Mail – is a free technology that is used to link a piece of email back to a domain.

DKIM plays a large role in addressing a flaw that has plagued email since the beginning. The flaw is: it's very difficult to identify legitimate email from fake email.

Due to the size, complexity, and depth of the installed base of email… which is used everywhere all the time by pretty much everyone… making email easy to identify is a major undertaking. The DMARC technical framework was designed to meet this challenge, and DKIM is one of two ways that DMARC uses to try to link a piece of email back to a domain.

To use DKIM, email servers are configured to attach special DKIM Signatures to the emails they send. These Signatures travel with the emails and can be verified along the way by the email servers that are doing the work of moving the emails toward their final destination. Each Signature contains all the information needed in order for an email server to verify that the Signature is real and that the content of the email that is attached to the Signature hasn't been modified.

In other words, DKIM allows people to attach a watermark of sorts to email so that email receivers can verify that the email actually came from a domain, and that the email hasn't been tampered with.

DKIM ends up being a nice technology in that DKIM signatures can survive forwarding and can render a degree of confidence that an email really did come from where it says it comes from.

This is about as much as can be said about DKIM without going into how it actually works.

DKIM creates a link between a domain and a piece of email through the DKIM Signature that gets attached to the piece of email. The DKIM Signature includes everything needed to verify that the signature is unique to the piece of email, that the piece of email hasn't been altered, and that the signature was created by an email server related to the domain under question.

To create a unique signature that only applies to a specific piece of email, the email server that adds the DKIM signature performs the following steps:

- The email server takes the piece of email and tries to boil off all the parts that are considered trivial. Things like whitespace, extra lines at the end of the email, how headers are folded. these are stripped out. DKIM has a few configuration options around how much boiling should be done, but since this is an overview, we'll just mention that the concept is called "canonicalization" and move on. What's left are the parts of the email that make it unique.
- The server then jams the email through a math function that yields a hash. This hash is not a breakfast food or a marijuana derivative, it's a long string that is unique to the email, sort of like a fingerprint. The server ends up making two hashes, one for the body of the email, and one for the headers of the email.
- The server has already been configured with a cryptographic key that is one-half of a key pair. This part of the key-pair is called the private key. Cryptography is a lot of fun, but because this is an overview of DKIM, we'll skip over all the fun crypto bits. Just be aware that the server can create

signatures that can be verified by anyone who has access to the other half of the key pair, which is called the public key.

- The server creates a cryptographic signature that covers the email. This is performed in a 2-step process that is quite clever, but unnecessary to go into during this overview.
- The resulting cryptographic signature is added to a DKIM-Signature header which is then inserted into the piece of email.
- The DKIM-Signature itself contains the body hash, the cryptographic signature, and information on where in the matching public key can be retrieved.
- The email is then released into the wild, sent on its way to eventually be delivered.

When someone or something gets a piece of email that contains a DKIM-Signature header, the email can be processed in a similar fashion to check the validity of the signature. The processing is like this:

- The trivial parts of the email are boiled off according to the options described in the DKIM-Signature header. This is the identical to what the creator of the signature did. The reason why this stuff is boiled off is because trivial changes to a piece of email can and do happen as email travels through email servers of various makes and models. We don't want those trivial changes to matter, so we strip them out.
- The email is jammed through the same math function to yield a hash for the body of the email and a hash for the headers of the email.
- Based on the information contained in the DKIM-Signature, the public key is retrieved from the DNS. "where can I find the servers that are responsible for accepting email on behalf of dmarcian.com?". You can query dmarcian.com using the DNS and you'll get back answers that have been put in place by the operators of dmarcian.com. In a similar vein, you can query dmarcian.com for the public key that is related to a specific DKIM-Signature.
- The public key is used to verify the cryptographic signature found in the DKIM-Signature.

If the verification process is successful, then the thing doing the checking knows that the signature was added by a piece of machinery that is directly related to the domain where the public key was located. This is how DKIM establishes a link between a domain and a piece of email.

The link that DKIM establishes between a domain and a piece of email is then used as input into things like anti-spam scanning, and more importantly, as input into DMARC-based checks.

If the verification process wasn't successful, then the receiving server must continue processing the email as usual without this link, as any number of things could be going on. The email might be real, but the message was modified too much in transit… thus causing the hashes to change, which in turn causes the cryptographic signature to not match, which means the DKIM-Signature ends up not verifying. OR, the email is fake and carrying a fake DKIM-Signature and is simply unwanted. It is difficult to attach meaning to the absence of the link that DKIM can provide.

Whether or not a link is identified, the result of the DKIM check is inserted into the email as part of the "Authentication-Results" header.

To conclude, DKIM is relatively sophisticated and has been in use for quite a few years. DKIM's utility has increased significantly since the advent of DMARC, as now domain owners have access to visibility into how their domain is being used that is in turn used to make sure all legitimate email streams are using DKIM.

Hopefully, this short overview provided insight into how DKIM works and how it's ability to create a link between a domain and a piece of email is important in the ongoing effort to make email easy to identify.