Workgroup: DKIM
Published: 8 March 2023
Intended Status: Informational
Expires: 9 September 2023
Authors: D. Crocker
Brandenburg InternetWorking

DKIM Replay: Problem Statement

Abstract

DomainKeys Identified Mail (DKIM, RFC6376) permits claiming some responsibility for a message by cryptographically associating a domain name with the message. For data covered by the cryptographic signature, this also enables detecting changes made during transit. DKIM survives basic email relaying. In a Replay Attack, a recipient of a DKIM-signed message re-posts the message to other recipients, while retaining the original, validating signature, and thereby leveraging the reputation of the original signer. This document discusses the resulting damage to email delivery, interoperability, and associated mail flows. A significant challenge to mitigating this problem is that it is difficult for receivers to differentiate between legitimate forwarding flows and a DKIM Replay Attack.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 9 September 2023.

Copyright Notice

carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction

> **[NOTE:]** This draft is based on the Problem Statement developed by Wei Chuan and others (including me) over some months. This version is offered as a refinement of that draft, with a tighter focus. Rather than being a 'separate' document, it should be treated as an aggressive edit of that draft. It has only my name on it, for now, since the revisions and decision to post it were only made by me, albeit with some advice from the WG Chairs.
>
> If this draft is adopted by the working group, I believe the document's authorship needs to revert to the list currently on Wei's version. /Dave

DKIM is a well-established email protocol RFC6376:

> DomainKeys Identified Mail (DKIM) permits a person, role, or organization to claim some responsibility for a message by associating a domain name RFC1034 with the message RFC5322, which they are authorized to use. This can be an author's organization, an operational relay, or one of their agents. Assertion of responsibility is validated through a cryptographic signature and by querying the Signer's domain directly to retrieve the appropriate public key.

1.1. The problem

The presence of a DKIM signature serves as a basis for developing an assessment of mail received, over time, using that signature. That assessment constitutes a reputation, which then serves to guide future handling of mail arriving with a DKIM signature for that domain name. The presence of a validated DKIM signature was designed to ensure that the developed reputation is the result of activity only by the domain owner, and not by other, independent parties. That is, it defines a 'clean' channel of behavior by the domain owner, with no 'noise' introduced by other actors.

A receiving filtering system contains a rich array of rules and heuristics for assessing email, for protecting users against spam, phishing, and other abuses. DKIM therefore provides an identity that this system can use for reputation assessment and prediction of future sender behavior.

During development of the DKIM specification, DKIM Replay was identified as only of hypothetical concern. However, that attack has become commonplace:

- Attackers create, obtain access, or compromise an account at a site with a high reputation.

- They send an email from that account to an external account also under their control.

- This single message is delivered to the attacker's mailbox, giving them an email with a valid DKIM signature, for a domain with high reputation.

- They then post the message to a new and large set of additional recipients.

Internet Mail permits sending a message to addresses that are not listed in the content To:, Cc: or Bcc: header fields. Although DKIM covers portions of the message content, and can cover these header fields, it does not cover the envelope addresses, used by the email transport service, for determining handling behaviors. So this message can then be replayed to arbitrary thousands or millions of other recipients, none of whom were specified by the original author.

That is, DKIM Replay takes a message with a valid DKIM signature, and distributes it widely to many additional recipients, without breaking the signature.

- Further, a message used in a Replay Attack has the same attributes as some types of legitimate mail. That is, an

individual, replayed message has no observable differences from a legitimate message.

Therefore, DKIM Replay is impossible to detect or prevent with current standards and practices. Simply put, email authentication does not distinguish benign re-posting flows from a DKIM Replay Attack.

ARC RFC8617 is a protocol to securely propagate authentication results seen by Mediators that re-post a message, such as mailing lists. It can be used to adjust DMARC RFC7489 validation as described in section 7.2.1. Because ARC is heavily based on DKIM it has the same "replay" issue as described in section 9.5.

1.2. Glossary

Modern email operation often involves many actors and many different actions. This section attempts to identify those relevant to Replay Attacks.

NOTE: This document is only Informative and omits the normative language defined in RFC2119. Mail architectural terminology that is used here is from RFC5598 and RFC5321.

RFC5598 defines mail interactions conceptually from three perspectives of activities, divided into three types of roles:

Users: This includes end-users, but also Mediators that re-post a message after delivery

Services (Message Handling Service - MHS): Moving a message from a single submission to its related delivery

Administrative (ADministrative Management Domain - ADMD): Covering independent operational scope, where functions of authorship, handling, and receiving can take place in any number of different ADMDs

Also, as noted in RFC5598, a given implementation might perform multiple roles.

It is useful to broadly identify participants in mail handling by functionality as defined in RFC5598 as:

*Mail Submission Agent (MSA)

*Mail Transmission Agent (MTA)

*Mail Delivery Agent (MDA)

In addition, a user interacts with the handling service via a:

*Mail User Agent (MUA).

The following is a subset of the Mail Handling Services defined in RFC5598 to be used in this document:

Originator: defined in Section 2.2.1. This is the first component of the MHS and works on behalf of the author to ensure the message is valid for transport; it then posts it to the a relay (MTA) that provides SMTP store-and-forward transfer. The Originator can DKIM sign the message on behalf of the author, although it is also possible that the author's system, or even the first MTA, does DKIM signing.

Alias: defined in Section 5.1. A type of Mediator user, operating in between a delivery and a following posting. The Alias replaces the original RCPT TO envelope recipient address but does not alter the content address field header fields. Often used for Auto-Forwarding.

ReSender: as defined in Section 5.2, is a type of Mediator user, like an Alias; however the ReSender updates the recipient address, and "splices" the destination header field and possibly other address fields as well.

Mailing Lists: defined in Section 5.3 is another Mediator; it receives a message and reposts it to the list's members; it mighty add list-specific header fields e.g. List-XYZ:,, might modify other contents, such as revising the Subject: field, or adding content to the body.

Receiver: defined in Section 2.2.4 is the last stop in the MHS, and works on behalf of the recipient to deliver the message to their inbox; it also might perform filtering.

Any of these actors, as well as those below, can add trace and operational header fields.

Modern email often includes additional services. Four that are relevant to DKIM Replay are:

Email Service Provider (ESP): Often called a Bulk Sender - An originating third-party service, acting as an agent of the author and sending to a list of recipients. They may DKIM sign as themselves and/or sign with the author's domain name.

Outbound Filtering Service: Rather than sending directly to recipients' servers, the Originator can route mail through a Filtering Service, to provide spam or data loss protection

services. This service may modify the message and can be administratively separate from the Originator.

Inbound Filtering Service: The Receiver can also route mail through a Filtering Service, to provide spam, malware and other anti-abuse protection services. Typically, this is done by listing the service in an DNS MX record. This service may modify the message and can be administratively separate from the Receiver.

The above services can use email authentication as defined in the following specifications:

DomainKeys Identified Mail (DKIM): Defined in RFC6376, with a cryptographic signature that typically survives basic relaying but can be broken when processed by a Mediator. Further, DKIM Replay is defined in RFC6376 section 8.6.

Sender Policy Framework (SPF): Defined in [RFC7208], is another form of message handling authentication that works in parallel to DKIM and might be relevant to the detection of a DKIM Replay Attack.

2. Mail Flow Scenarios

The following section categorizes the different mail flows by a functional description, email authentication and recipient email header fields.

2.1. Basic types of flows

Direct delivery: In this case, email travels directly from the author's ADMD or the ADMD of their agent -- to the recipient's ADMD or their agent. That is, for origination and reception, any interesting creation or modification is done by agreement with either the author or the recipient. As such, these cases should have authentication that succeeds.

In this type of flow, SPF is expected to validate.

A DKIM Replay Attack uses a single message, sent through Direct delivery, and repurposes it.

Indirect Delivery This is mail involving a Mediator, producing a sequence of submission/delivery segments. While not required, the Mediator is typically viewed as being in an ADMD that is independent of the author's ADMD and independent of the recipient's ADMD.

2.2. Direct examples

ESP :

An ESP is authorized to act on behalf of the author and will originate messages given a message body and a list of recipients, sending a different message to each recipient. Content address fields are typically restricted to just the address of that copy's recipient. The mail that is sent is typically 'direct', but the ESP cannot control whether an address refers to an alias or mailing list, or the like. So, the message might become indirect, before reaching the final recipient.

The bulk nature of ESP activity means that it can look the same as DKIM Replay traffic.

Outbound filtering If the Author's domain has an SPF record that does not list this filtering service, SPF validation for the author's domain will fail. However, the ESP might produce an SPF record of their own and use their own SMTP MAIL FROM (return) address.

Inbound filtering : Typically, an inbound filtering service will add the results of its analysis to the message. It might make other modifications to the message.

2.3. Indirect Examples

Indirect mail flows break SPF validation, unless the Mediator is listed in the SPF record. This is almost never the case.

Mailing List: The modifications done by a mailing list especially to the Subject: header field and the body of the message - nearly always break any existing DKIM signatures.

Alias (e.g., Auto-forwarder): Typically, the envelope return (MAIL FROM) address is replaced, to be something related to the forwarder. A resender might add trace headers, but typically does not modify the recipients or the message body.

3. DKIM Replay

3.1. Scenario

A spammer will find a mailbox provider with a high reputation and that signs their message with DKIM. The spammer sends a message with spam content from there to a mailbox the spammer controls. This received message is sometimes updated with additional header fields such as To: and Subject: that do not damage the existing DKIM signature, if those fields were not covered by the DKIM signature. The resulting message is then sent at scale to target recipients. Because the message signature is for a domain name with a high

reputation, the message with spam content is more likely to get through to the inbox. This is an example of a spam classification false negative incorrectly assessing spam to not be spam.

When large amounts of such spam are sent to a single mailbox provider -- or through a filtering service with access to data across multiple mailbox providers -- the operator's filtering engine will eventually react by dropping the reputation of the original DKIM signer. Benign mail from the signer's domain then starts to go to the spam folder. For the benign mail, this is an example of a spam classification false positive.

In both cases, mail that is potentially wanted by the recipient becomes much harder to find, reducing its utility to the recipient (and the author.) In the first case, the wanted mail is mixed with potentially large quantities of spam. In the second case, the wanted mail is put in the spam folder.

3.2. Direct Flows

Legitimate mail might have a valid DKIM signature and no associated SPF record.

So might a Replay attack.

3.3. Indirect Flows

Example benign indirect flows are outbound and inbound gateway, mailing lists, and forwarders. This legitimate mail might have a valid DKIM signature, and SPF validation that is not aligned with the content From:

So might a Replay attack.

4. Replay technical characteristics

A message that has been replayed will typically show these characteristics:

- Original DKIM signature still validates
- Content covered by that signature is unchanged
- Received: header fields might be different from the original, or at least have ones that are added
- SMTP Envelope RCTP-TO address will be different
- SMTP MAIL-FROM might be different

*Replayed mail will typically be sent in very large volume

*The original SPF will typically not validate; however if the MAIL-FROM has been changed to a an address controlled by the spammer, SPF might validate.

5. Basic solution space

[NOTE:] The chairs have expressed a desire for the Problem document to refrain from discussing the solution space. Since this document introduces the reader to the topic of DKIM Replay Attacks, and might receive wide circulation, I think there can be some benefit in trying to head off simplistic thinking about solutions. This section is abbreviated from the other draft, to merely highlight some solution issues. The section is easily removed, of course... [Dave]

As can be seen from the above discussion, there is no straightforward way to detect DKIM Replay for an individual message, and possibly nothing completely reliable even in the aggregate. The challenge, then, is to look for passive analysis that might provide a good heuristic, as well as active measures by the author's system to add protections.

Here are some potential solutions to the problem, and their pros and cons:

Include the SMTP RCPT-TO address in the DKIM signature:

Since this information is different in the Replay, than it was in the original sending, locking it into the signature will make validation fail, if the value has been changed.

-This avoids Replay to destination addresses not anticipated by the DKIM signer.

-Indirect flows will fail, since forwarding involves rewriting the ENVELOPE-TO; however they already typically fail.

-This will detect DKIM Replays, but not distinguish them from all other forwarding.

-If a message has more than one addressee, should the signature cover all of them, or does this require sending one message per addressee? If it covers all of them, note that they might be on different systems, so that upon arrival, the RCPT-TO list will not include all of the original addresses

Cache known DKIM signatures, to support aggregate analysis:

- Since the same signature is being replayed many times, this might allow a receiving site with many mailboxes to detect whether a message is part of a DKIM Replay set, and to then suppress it.

- Mailing list traffic, aliases, and the like might also show up as duplicates. So this is only an heuristic, and might produce false positives.

Strip DKIM signatures on mailbox delivery:

- Messages delivered to a mailbox are not able to be replayed any more.

- Has no effect when the receiving platform is collaborating with the bad actor, as the attacker would just avoid stripping the header fields.

Shorten DKIM signature key lifetime:

- If the key is no longer available through the DNS, the signature will no longer validate

- Unfortunately, bad actors are quite good at taking action very quickly, and there is a limit to how much the window can be shortened, if the key is to have any utility for legitimate mail

Add a per-hop signature, specifying the destination domain for the next hop:

- Messages with this kind of signature cannot be replayed down a different pathway, since the destination won't match.

- Requires every site along the path to support this spec, and to detect whether the next stop is making a commitment to follow the spec.

- If email goes to a site that does not support this behavior, traversing a naive forwarder remains indistinguishable from Replay.

- The time needed to change a global infrastructure such as email, to fully support a capability like this in every MTA is essentially infinite; therefore use of this approach must be narrowly tailored to scenarios that will adopt it and garner substantial benefit from it.

6. Security Considerations

Author's Address

Dave Crocker
Brandenburg InternetWorking
675 Spruce Drive
Sunnyvale, CA 94086
United States of America

Phone: +1.408.246.8253
Email: dcrocker@bbiw.net