

Authenticated Received Chain Overview



DMARC.org



Introduction to DMARC.org

The mission of DMARC.org is to promote the use of DMARC and related email authentication technologies to reduce fraudulent email, in a way that can be sustained at Internet scale. This overall goal is met by educating individuals and organizations through a combination of articles, tutorials, and presentations.

For more information, please visit <https://dmarc.org>

DMARC.org is an initiative of the non-profit Trusted Domain Project (TDP). For more about TDP, please visit <http://trusteddomain.org>

The contents of this presentation are released under the [Creative Commons Attribution-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-sa/4.0/) (CC BY-SA).





Introduction to DMARC.org

The work of DMARC.org is made possible through the generous support of these companies:

Sponsors



Supporters



Background





What Was Done Before ARC?

- Previous work had been done on a header to convey authentication results between domains (ADMDs)
- Original Authentication Results (OAR) was published as an Internet Draft in February 2012
- Assumes trust between ADMDs – not widely used
- Some large enterprises used it internally
- Might address issue some domains using DMARC experienced with *indirect mailflows*

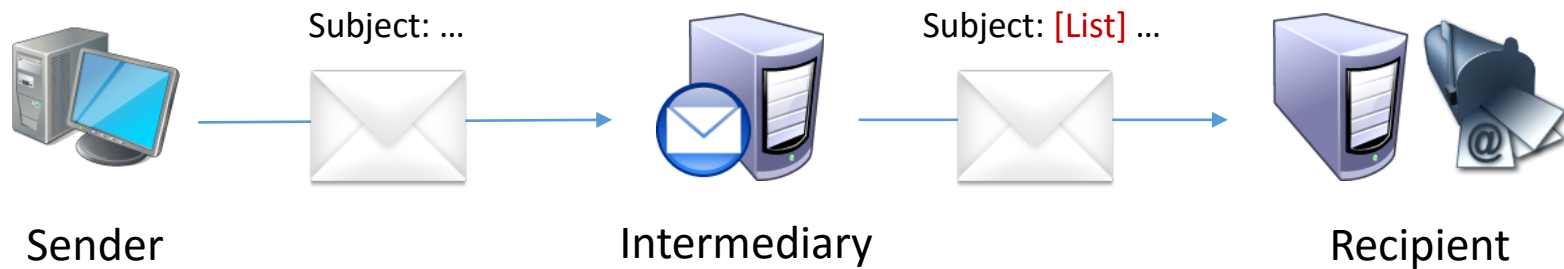


Connection Between ARC and DMARC

- Domains with strict DMARC policies (`p=reject`) may see legitimate messages blocked if they go through *indirect mailflows* such as mailing lists, forwarding, or filtering services
- Forwarding causes SPF to fail even if origin was legit
- Forwarders often alter messages, breaking DKIM
 - Disclaimers and footers
 - Virus scan results
 - Removed attachments
 - Mailing list subject tags



Example of an Indirect Mailflow



- Intermediary sends the message from a new IP address, causing SPF to fail to verify for Sender's domain
- Intermediary changes the message contents, causing Sender's DKIM signature to fail to verify



Why Was ARC Created?

- In April 2014, AOL and Yahoo published a `p=reject` DMARC policy for their customer-use domains
- While this affected less than 1% of their customers' email, there was significant disruption for many users of indirect mailflows
- Ad hoc working group formed to adapt OAR to address these disruptions of indirect mailflows
- Significant changes required for a general solution, so a new name was chosen



Design Decisions for ARC

- Originator of message makes no changes
- Convey the `Authentication-Results`: content intact from the first ARC intermediary forward
- Allow for multiple “hops” in the indirect mailflow
- ARC headers can be verified at each hop
- Work at Internet scale
- Define ARC independently of DMARC if possible



Design Decisions for ARC

- Message recipient seeing an authentication failure under DMARC may choose to check ARC headers
- If ARC headers are intact, they can see and validate `Authentication-Results`: content reported by the first ARC participant
- Depending on reputation of intermediary/-ies and results, message recipient *may* choose to use ARC information as basis for a “local override” of authentication checks like DMARC



What Does ARC Do?

- Intact ARC chains give you:
 - DKIM, DMARC and SPF results as seen by first “hop”
 - Signatures showing these results were conveyed intact
 - Signatures from participating intermediaries can be reliably linked to their domain name
- Allows intermediaries to alter message with attribution
- ARC can provide data on intermediaries to a reputation system tracking their behavior



What Doesn't ARC Do?

- Does not say anything about “trustworthiness” of the message sender or intermediaries
- Says nothing about the contents of the message
- Intermediaries might still inject bad content
- Intermediaries might remove some or all ARC headers

Implementation





Three New Header Fields

- ARC-Authentication-Results: (AAR)
Archived copy of Authentication-Results:
- ARC-Seal: (AS)
Includes some tags and a DKIM-style signature of any preceding ARC headers/sets
- ARC-Message-Signature: (AMS)
A DKIM-style signature of the entire message except ARC-Seal: headers



ARC-Authentication-Results: (AAR)

- Copy of the contents of the locally generated `Authentication-Results: header`
- One addition – the `i=` tag is prepended, containing a sequence number for the current set of ARC headers



ARC-Message-Signature: (AMS)

- A modified DKIM signature – leverages existing libraries
- **i=** tag is different – under ARC, a sequence number for ARC header sets
- **v=** tag is missing in ARC
- Should not be usable as a DKIM signature in a replay attack



ARC-Seal: (AS)

- Populated with *key=value* pairs
- **b=** is a signature of all ARC headers, no non-ARC hdrs
- **a=/d=/s=** fields match the corresponding DKIM tags
 - Same key format and DNS records as for DKIM
 - Can use your DKIM keys for ARC
 - Can use separate keys per local policy or preference
- **cv=** indicates whether ARC chain validated as received by the reporting intermediary
- **i=** tag is a sequence number for ARC header sets



Order of Insertion

- `Authentication-Results`: content is copied into a new `ARC-Authentication-Results` header, prefixed to the message
- `ARC-Message-Signature`: is calculated for message, including newest AAR header, and prefixed to the message
 - Must not include any `ARC-Seal` headers
- `ARC-Seal`: is calculated and prefixed
- ARC headers prefixed per common practice, but order of appearance is not critical for validation



The `i=` Sequence Number

The `i=` sequence tag is used to order the ARC headers for various operations

- Allows multiple ARC header sets to be grouped easily and correctly
- Eliminates reliance on the order of headers being inserted – or not being altered
- Compare with order of insertion of various authentication, content scanning, or `Received:` headers



What Constitutes A Valid ARC Chain

Method used by each participant to determine the **cv=** value in their `ARC-Seal`:

- All `ARC-Seal`: headers must validate
- The **cv=** value for those AS headers must be Pass
- The most recent `ARC-Message-Signature`: (highest **i=** value) must validate



When Would I Insert ARC Headers?

- When a message is subject to handling that will knowingly break existing DKIM signatures
 - Inserting `Subject: tags`
 - Appending disclaimers and footers
 - Stripping attachments
 - Content-encoding changes
- When the message crosses a trust boundary, which might occur exiting an ADMD
 - Sometimes within, e.g. a multi-department or multi-entity enterprise

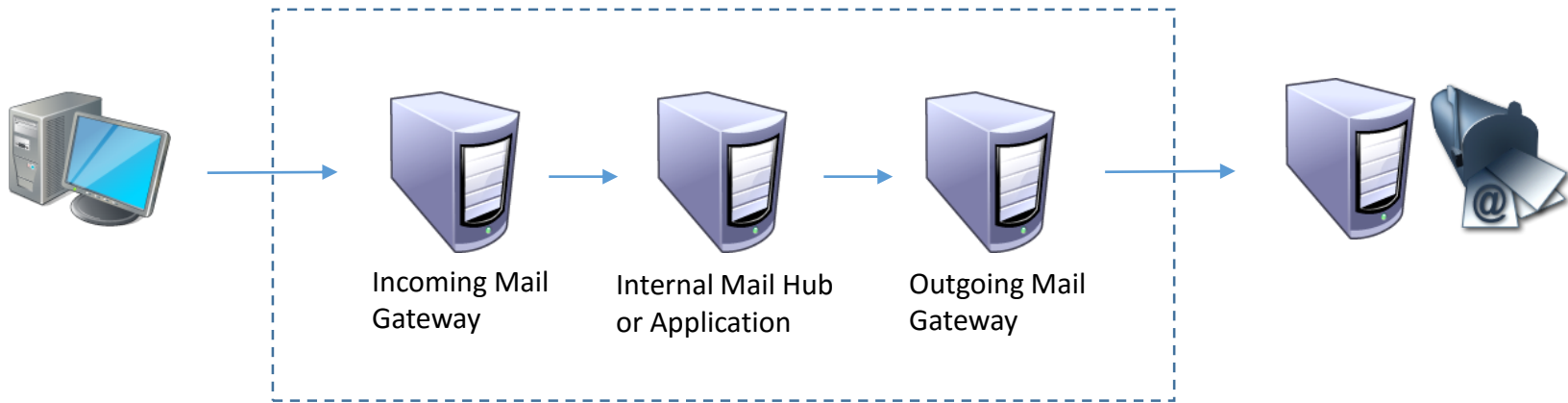


When Would I Insert ARC Headers?

Sender

Intermediary

Recipient



Incoming Gateway validates
ARC, DKIM, DMARC, SPF
normally

Outgoing Gateway inserts
DKIM signature, inserts ARC
headers

Different organizations will have different configurations, but still check ARC on inbound messages and insert ARC when messages are outbound





When Wouldn't I Insert ARC Headers?

- When the message will be delivered to a mailbox within the local organization (ADMD)
- ARC builds a verifiable chain of intermediate message handlers – anonymous remailers might not find this desirable...



What Do ARC Headers Look Like?

Origin

Basic message headers,
DKIM-Signature

DKIM-Sig:
To:
From:
Subject:
.
.
.

Mailing List

Checks auth; Adds
Auth-Results:, DKIM-
Signature, ARC
headers, Subject tag

ARC-Seal: i=1
ARC-Msg-Sig: i=1
ARC-Auth-Res: i=1
DKIM-Sig:
Auth-Results:
DKIM-Sig:
To:
From:
Subject: [List]
.
.
.

Hop 2

Checks auth; Adds
Auth-Results:, DKIM-
Signature, ARC
headers

ARC-Seal: i=2
ARC-Msg-Sig: i=2
ARC-Auth-Res: i=2
DKIM-Sig:
Auth-Results:
ARC-Seal: i=1
ARC-Msg-Sig: i=1
ARC-Auth-Res: i=1
DKIM-Sig:
Auth-Results:
DKIM-Sig:
To:
From:
Subject: [List]
.

Destination

Checks auth; Unpacks
ARC headers; adds
Auth-Results:

Auth-Results: arc=...
ARC-Seal: i=2
ARC-Msg-Sig: i=2
ARC-Auth-Res: i=2
DKIM-Sig:
Auth-Results:
ARC-Seal: i=1
ARC-Msg-Sig: i=1
ARC-Auth-Res: i=1
DKIM-Sig:
Auth-Results:
DKIM-Sig:
To:
From:
Subject: [List]





How Are ARC Verdicts Shown?

- `arc=pass` **or** `arc=fail` may be inserted into `Authentication-Results: headers`
- DMARC-aware receivers who validate ARC results should include ARC information in DMARC aggregate reports, `local_policy` section:

```
<reason>
  <type>local_policy</type>
  <comment>arc=pass ams=d1.example d=d1.example,d1.example</comment>
</reason>
```

- `ams=` is the **d=** domain from the last AMS header
- `d=` is the list of **d=** domains from validated `ARC-Seal`:



Summary





Benefits of ARC

Sender/Intermediary Benefits

- Allow intermediaries to continue or resume traditional `From:` semantics, message modifications
- Allow more senders to adopt `p=reject` DMARC policies, blocking fraudulent messages
- Improves overall deliverability

Receiver Benefits

- Allow more receivers to enforce DMARC policies
- More data for reputation systems
- Allow more mailbox providers to publish `p=reject` policies on their customer-facing domains





ARC Timeline

- October 2015:
 - Announcement at M³AAWG 35 in Atlanta
 - Draft specification and usage doc published as IETF Internet-Drafts
- Fall 2015 – Spring 2016:
 - AOL, GMail, and OpenARC implementations developed
- February 2016
 - Interoperability event #1
- March-April 2016
 - Updates to the specification
- May 2016
 - Interoperability event #2
- June-July 2016
 - Interoperability event #3





ARC Resources

- Website for latest ARC news:
<http://arc-spec.org>
- Mailing List for discussion of ARC:
<http://lists.dmarc.org/mailman/listinfo/arc-discuss>
- Specification, current draft:
<https://tools.ietf.org/html/draft-andersen-arc-04>
- Usage Guidelines, current draft:
<https://tools.ietf.org/html/draft-jones-arc-usage-01>



Questions

