JMAP extensions for S/MIME signature verification

draft-ietf-jmap-smime-10

Alexey Melnikov <alexey.melnikov@isode.com>

Changes in -10

- Addressed ART Directorate, GenArt, SecDir review comments, IESG reviews. In particular:
- Unclear/missing references
- Clarified trust model
- Expanded Security Considerations section
- smimeStatus attribute extensibility
- Added new attributes to Email/query. For example to search for all messages with valid S/MIME signature
- Corrected examples

Feedback from IESG

- 10 minutes caching time
  - Several people asked about why S/MIME result caching is 10 minutes
  - The point of caching is to protect JMAP server from DoS attacks, as checking all signatures, certificate validity is CPU/network expensive
  - Certificate expiration is not going to happen very often, so caching can probably be done for days
  - However certificate revocation can happen at any time. So would it be reasonable to wait for 1 day, for example?

Feedback from IESG

- IANA registry for smimeStatus values?
  - smimeStatus is extensible to allow JMAP S/MIME extensions such as automatic S/MIME decryption
  - I can’t think of other cases, but it doesn’t mean they don’t exist
  - New values affect how some dynamic attributes are evaluated, for example “all messages with valid signature” Email/query attribute
- 2 possible solutions:
  - Just add an IANA extension. Probably with the Expert Review policy.
  - Don’t create the IANA registry, but add known values for automatic decryption (e.g. encrypted+signed/verified and encrypted+signed/failed), saying that they are reserved for future use. The expectation is that they would be described by a future draft.

Feedback from IESG

- Add hasVerifiedAtDeliverySmime Email/query attribute?
  - As we have an Email/get attribute for requesting S/MIME status at delivery time, it makes sense to be able to search for “all messages that had valid S/MIME status at delivery”
- Thoughts?

Feedback from IESG

- Is "signed" smimeStatus attribute value sensible if "S/MIME status at Delivery" attribute is required to be implemented?
  - “signed” means that the message has an S/MIME signature, but its validity hasn’t been verified yet. This is an optimization to allow a JMAP client to display “the message has S/MIME signature” in the message list, without the need for the server to spend resources to verify it. **Clarify this in the document.**
  - “S/MIME status at Delivery” actually doesn’t need to be verified at delivery time. (**This can be clarified in the document.**) It can be calculated when explicitly requested, as the delivery time is known from the message.
  - Proposal:
    - keep “signed”
    - don’t have a separate capability for “S/MIME status at Delivery” - incremental cost of implementing it is minimal, but it probably will not be implemented by itself.

Feedback from IESG

- Ben Kaduk wrote: “The immutability of smimeStatusAtDelivery might have unfortunate interactions with trust anchor changes (additions, mostly) on the server, since the state could be locked into invalid even if the current TA set would have verified it at the time of delivery.”
- Check that "server-set" doesn’t mean "immutable".
  - If it does mean immutable, then maybe just note this in the Security Considerations section.
  - If it doesn’t mean immutability, then mention this as an example of when the attribute status might change.

Feedback from IESG

- Roman: Just reference [RFC8551] and [RFC8550] in definitions of signed/verified and signed/failed?
  - **Check the above RFCs**. Make it clear that From/Sender header field mismatch against any email from the sender's certificate should result in signed/failed, unless there is a privately implemented server configuration that maps or authorizes mismatching email addresses
- **Check the above RFCs**. Do we also need to remind people about doing certificate revocation checks?
- Am I trying to fix possible defects in S/MIME RFCs?

Next steps

- Address comments as per discussion in this session and post a new draft.

- Ask IESG to approve for publication as an RFC.