draft-peterson-dispatch-rtpsec

Jon Peterson

DISPATCH WG

IETF 95 (Buenos Aires)

Media Confidentiality with SIP

- Goal: show practices for establishing media confidentiality for sessions set up with SIP
  - Targeting BCP status
- Why?
  - PERPASS (RFC7258)
  - Hopefully influence implementation and/or policy
    - More prescriptive than descriptive, like PERPASS itself
  - Also, as we put this together, we will identify gaps
    - Story here is pretty good, but there are limitations

What Does the Draft Say?

- Divides into two confidentiality methods
  - **Comprehensive** security
    - Use STIR (successor to RFC4474)
    - STIR object signs media fingerprints in SDP
      - Binds keys to the SIP-layer identities signed by STIR
  - **Opportunistic** security
    - Use draft-johnston-dispatch-osrtp, basically
      - Offer AVP rather than SAVP, but provide key info in SDP
    - This document doesn't replace OSRTP, points to it

Does STIR Work for This?

- STIR revises the RFC4474 SIP Identity header
  - Scope narrowed to prevent impersonation for a set of specific threats (e.g. robocalling)
  - MitM protections not in scope
    - However, does provide the mky field as a hook
- Provides an **authentication service** abstraction that signs SIP requests
  - Can be implemented at endpoints or intermediaries
    - Signed at intermediaries, media protection is not E2E
    - Fine for STIR's threat model, not great for media sec
  - Verifiers have no real way to tell if the sig is E2E

Opportunistic STIR?

- Could STIR sign requests without vouching for the originator's identity?
  - Added some "don't rule this out" text to latest rfc4474bis
  - Would provide an auth service sig over the key fingerprints/hashes in SDP without identity
  - Ideally implemented in endpoint auth services
    - They might in turn use self-signed keys, even
    - Can be supplied in addition to "real" Identity header
- Does it add any real benefit over OSRTP?
  - Shows that media keys have not been tampered with in transit (at least since they were signed…)
    - Basically with TOFU trust of auth services

Connected Identity

- STIR (and original RFC4474) only signs SIP requests
  - No signatures over SIP responses
- Elwell's RFC4916 patches this
  - UPDATE in the backwards direction sent after a PRACK or a 2xx
  - Or re-INVITE in an established dialog
  - RFC4916 lets the UAS alter To/From to show who you actually connected to
  - Also allows SDP for early media in these requests
- RFC4916 would need some post-STIR tweaks
  - Basically, though, this is a blueprint for signing SDP in the backwards direction for media confidentiality

Media Security

- OSRTP allows DTLS-SRTP, MIKEY, ZRTP, sdesc
- People defend MIKEY for some corner cases
- This might be a good place to deprecate sdesc entirely
- Ultimately, need some MTI for a BCP
  - That would surely be DTLS-SRTP
  - Provide options for others, including ZRTP
  - Is this a good direction? Mic check?
- This BCP and OSRTP should be aligned on these

Going Forward

- Reasonable ideas?
- Are these the best practices?
  - Can’t help but notice some are still in development
- Where to do the work?
  - Any tweaks to RFC4916 could be dispatched to SIPCORE
  - Opportunistic STIR could be a STIR draft
  - Would this BCP need its own WG? Should it be tacked on to STIR’s charter? AD sponsored? Other thoughts?