KITTEN WG
January 2023 Interim

Chair:
Alexey Melnikov <alexey.melnikov@isode.com>

Note Well

- This is a reminder of IETF policies in effect on various topics such as patents or code of conduct. It is only meant to point you in the right direction. Exceptions may apply. The IETF's patent policy and the definition of an IETF "contribution" and "participation" are set forth in BCP 79; please read it carefully.

- As a reminder:

    - By participating in the IETF, you agree to follow IETF processes and policies.

    - If you are aware that any IETF contribution is covered by patents or patent applications that are owned or controlled by you or your sponsor, you must disclose that fact, or not participate in the discussion.

    - As a participant in or attendee to any IETF activity you acknowledge that written, audio, video, and photographic records of meetings may be made public.

    - Personal information that you provide to IETF will be handled in accordance with the IETF Privacy Statement.

    - As a participant or attendee, you agree to work respectfully with other participants; please contact the ombudsteam (https://www.ietf.org/contact/ombudsteam/) if you have questions or concerns about this.

Note Well
(continued)

- Definitive information is in the documents listed below and other IETF BCPs. For advice, please talk to WG chairs or ADs:
  - BCP 9 (Internet Standards Process)
  - BCP 25 (Working Group processes)
  - BCP 25 (Anti-Harassment Procedures)
  - BCP 54 (Code of Conduct)
  - BCP 78 (Copyright)
  - BCP 79 (Patents, Participation)
  - https://www.ietf.org/privacy-policy/ (Privacy Policy)

IETF Code Of Conduct Guidelines RFC 7154

- Treat colleagues with respect
- Speak slowly and limit the use of slang
- Dispute ideas by using reasoned argument
- Use best engineering judgment
- Find the best solution for the whole Internet
- Contribute to the ongoing work of the group and the IETF

Administrivia

- This session is being recorded
- Meetecho:
  - https://meetings.conf.meetecho.com/interim/?short=6011e6f6-2503-40db-b99b-68f88c85a0e9
- Shared note taking:
  - https://notes.ietf.org/notes-interim-kitten-jan-2023
- ***Note taker***?

Agenda

- Proposed SASL2 features above SASL1 [RFC4422]:
  - Upgrade ("password change") tasks
  - Two factor authentication (2FA) tasks
  - Channel binding advertisement
  - Shared key derivation after successful authentication
- SCRAM 2FA draft update (draft-ietf-kitten-scram-2fa-02)
- Fast reauthentication SASL mechanism (draft-schmaus-kitten-sasl-ht-09)
- OPAQUE SASL mechanism status update (draft-reitzenstein-kitten-opaque-01)

XSF SASL2 proposal and related documents

- **Extensible SASL Profile: https://dyn.eightysoft.de/final/xep-0388.html**
- **SASL2 Upgrade Tasks: https://dyn.eightysoft.de/final/xep-scram-upgrade.html**
- **SASL Channel-Binding Type Capability: https://dyn.eightysoft.de/final/xep-0440.html**

- Tasks allow servers to request an extra authentication related operations
  - Upgrade ("password change") tasks
    - Help with secure hash migration of accounts after a successful authentication, when plaintext password is not stored
  - Two factor authentication (2FA) tasks
- Channel bind advertisement
- Shared key derivation after successful authentication
- **More details in the following presentation**

XSF SASL2 presentation

- **Matthew Wild to present**

Shared key derivation after successful authentication

- **Rick van Rein to present**

draft-ietf-kitten-scram-2fa-02 status update

- 2 2FA mechanisms defined in the draft now:
  - TOTP (6 digits)
  - **FIDO2 CTAP1 - works for USB, NFC and BLE**

- Also an extension to return reauthentication token (using o= attribute)
  - This should be compatible with draft-schmaus-kitten-sasl-ht-09, without requiring any SASL2 features!
    - I.e. draft-schmaus-kitten-sasl-ht can be used with any SASL mechanism that has a way to create reauthentication token.

Fast reauthentication

- Applications need to reauthenticate frequently:
  - They get disconnected and want to provide good user experience by transparently reconnecting
  - They sometimes need to create more than 1 parallel connection
    - Asking user to do 2FA on each connection is ruining user experience

- Advantages of defining a fast reathentication SASL mechanism
  - Allows to bypass 2FA
  - Also less CPU intensive/fewer round trips than the initial authentication using SCRAM or OPAQUE mechanism

draft-schmaus-kitten-sasl-ht-09

- Fast reauthentication SASL mechanism
- Florian Schmaus to say a few words

draft-reitzenstein-kitten-opaque-01

- OPAQUE SASL mechanism based on draft-irtf-cfrg-opaque
  - Reuses syntactic structure (and many features) of SCRAM, which makes it both a SASL and GSSAPI mechanism (based on GS2 framework from RFC 5801)
  - 2FA extensions from draft-ietf-kitten-scram-2fa-02 would also work for this SASL mechanism

- Nadja?

What is next?

- Alexey's proposal:
  - I prefers to add 2FA support without waiting for SASL2 to be fully specified and deployed, even though SASL2 features sound great!
    - Is draft-ietf-kitten-scram-2fa-02 on the right track?
  - Start adoption call on draft-schmaus-kitten-sasl-ht in KITTEN WG
    - Again, I prefer this to be useful with both SASL1 and SASL2
  - Cherry pick the most supported bits of SASL2 proposal and write a KITTEN WG draft?
    - I would like this to be usable in SMTP/IMAP, not just XMPP
    - We can start with the bare bone draft (e.g. that includes "tasks") and add other features if there is rough consensus to add them