redhat.

NSS and TLS 1.3

ICMC 2018

Robert Relyea
Principle Programmer
May 11, 2018

Synopsis

- Who am I
- NSS and Standards Compliance (read FIPS)
- TLS 1.3 vs. TLS 1.2 in NSS
- TLS 1.3 challenges in NSS
- When will we see all this?
- Questions

Who am I?

- Long Time NSS developer (since 1996)
- Worked for Netscape, iPlanet (aka Sun/Nescape Alliance), AOL, and Red Hat
- Currently part of the Red Hat Crypto Team responsible for NSS, openSSL, gnuTLS and indirectly responsible for all the crypto in Red Hat.
- OASIS PKCS #11 co-chair

NSS and Standards Compliance

- PKCS #11 is the FIPS boundary.
  - AES/GCM presents a difficulty to the PKCS #11 2.0 interface. This has been fixed in PKCS #11 3.0.
- In FIPS mode, keys are ‘locked’ to the token and can not be removed in the clear.
  - The SSL implementation has no access to key data, and implements no crypto or key exchanges.
  - TLS Mac and TLS KDF are both implemented in the PKCS #11 module, not the SSL code.
- Algorithms are not ‘locked’ in software. They are ‘locked’ in the security policy.

TLS 1.2 vs. TLS 1.3 in NSS

- The engine is quite different. We rewrote the handshake handling state machine to handle TLS 1.3.
- This lead to finally dropping SSL 2.0 support altogether, and lead to starting the process of removing SSL 3.0 support.
- TLS 1.3 uses a different KDF. Already had support for HKDF through a private PKCS #11 NSS Mechanism.
- Basically everything but the record layer was changed in TLS 1.3

TLS 1.3 challenges

- The implementation of TLS 1.3 was done by Mozilla, mostly by people like Eric Rescorla and Martin Thompson.
- The TLS state machine was rewritten.
- The TLS 1.3 KDF used an existing function (HKDF), however it did things like pass the key in as the salt parameter, which causes problems with the HDKF implementation within NSS.
- Rollout issues.
  - We wanted people to be able to try out TLS 1.3 draft, but people would enable it by accident (when the enabled the highest version of TLS in their software).
  - Same issue with new TLS cipher suites.

When will we see all this?

- The NSS upstream code has the latest draft. It will go live once the TLS spec goes final.
- Daft 28 was posted March 30. We have every reason to believe it will be the final draft once IEFT dot their ‘i’s and cross all their ‘t’s.
- The current PKCS #11 spec is sufficient for the current implementation.
- PKCS #11 3.0 with updated support for GCM should go live by the year’s end.
- PKCS #11 3.0 is ‘closed’ and does not have HKDF.
  - HKDF draft that deals with things like using another key for the salt will be available once PKCS #11 3.0 goes out for review.

redhat.

Questions

plus.google.com/+RedHat

facebook.com/redhatinc

linkedin.com/company/red-hat

twitter.com/RedHatNews

youtube.com/user/RedHatVideos