draft-friel-tls-eap-dpp-05

Dan Harkins & Owen Friel

EMU WG

IETF 114, Philadelphia, Pennsylvania

Summary

- How to solve the on-boarding *Catch-22* for wired devices w/o a rich UI
- Reuse Wi-Fi alliance Easy Connect / Device Provisioning Profile (DPP) bootstrap mechanism for wired bootstrap
- Reuse DPP EC bootstrap key pair and formats (e.g. QR code)
- Provides mutual authentication between bootstrapping client and server that knows client's bootstrap public key
- Use RFC 5869 HKDF to derive PSK from bootstrap key
- Use RFC 8773 Cert Based Auth with External PSK
- Use RFC 7250 TLS with raw public key using bootstrapping key
- Use draft-ietf-tls-external-psk-importer to import derived PSK
- No new TLS extensions or changes required

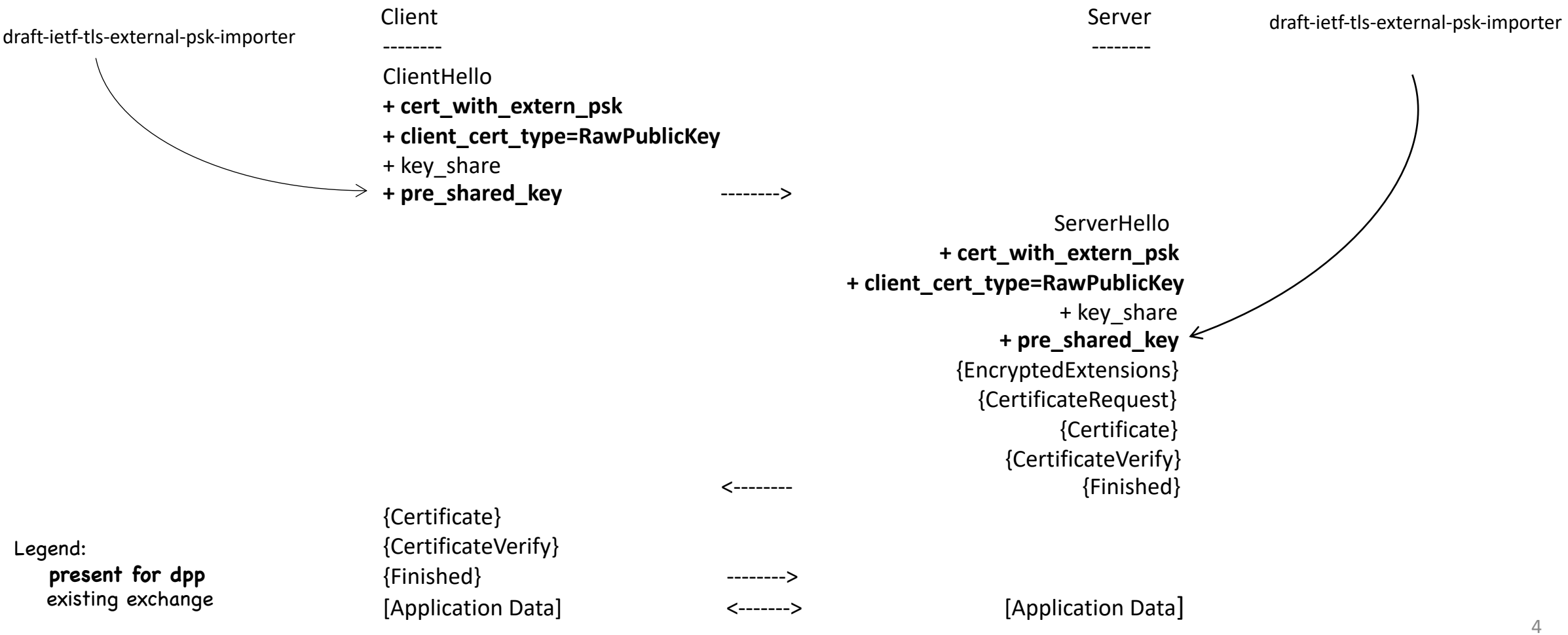
Changes since IETF 113 draft-04

- From: draft-group-tls-extensible-psks
  - Work current onhold
  - Not yet published as an IETF draft
- To: draft-ietf-tls-external-psk-importer
  - Specify how to convert DPP bootstrapping key to ImportedIdentity

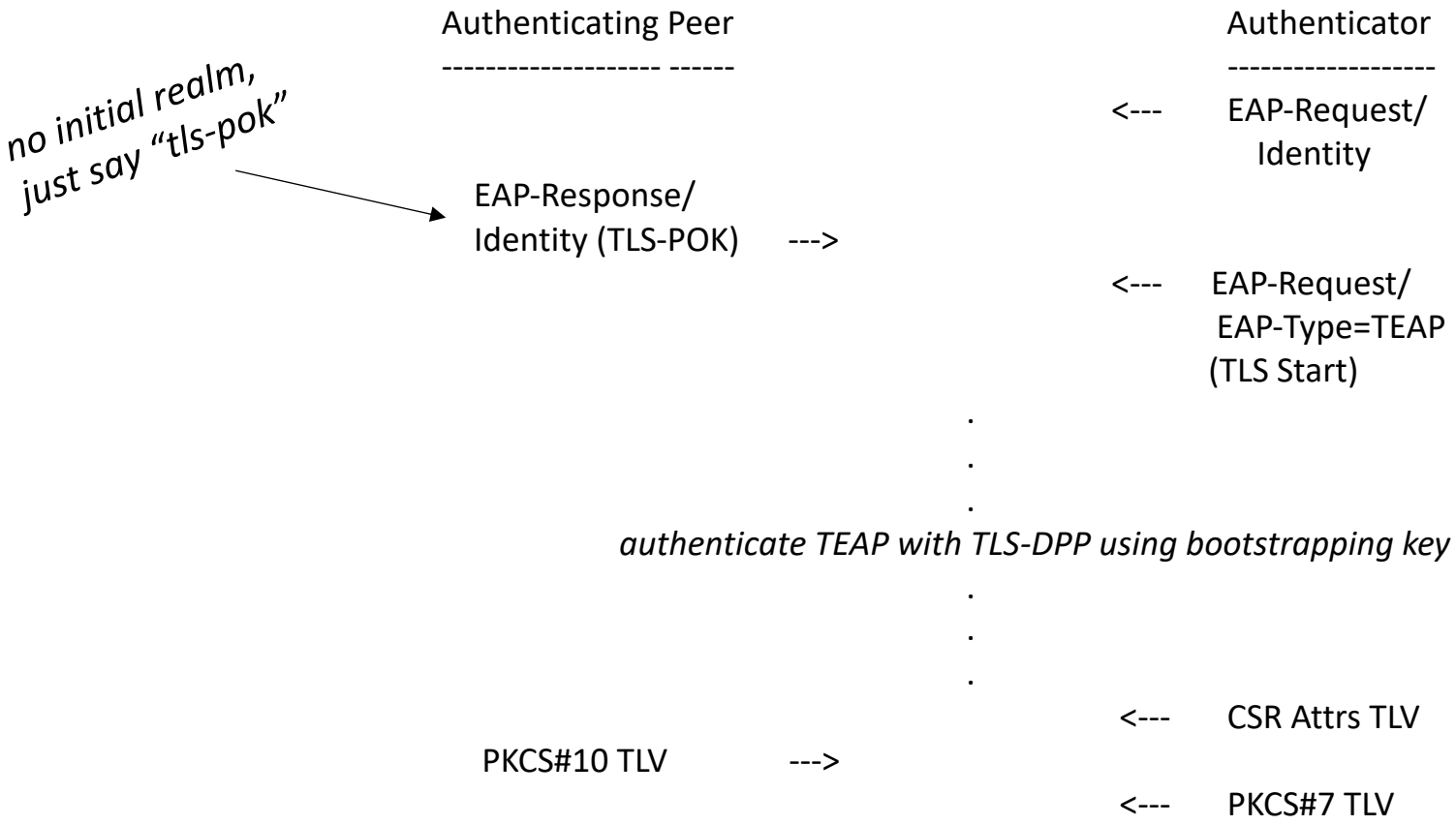
```
epsk   = HKDF-Expand(HKDF-Extract(<>, bskey),
                     "tls13-imported-bsk", L)
epskid = HKDF-Expand(HKDF-Extract(<>, bskey),
                     "tls13-bspsk-identity", L)
where:
  - epsk is the EPSK Base Key
  - epskid is the EPSK External Identity
  - <> is a NULL salt
  - bskey is the DER-encoded ASN.1 subjectPublicKeyInfo
    representation of the BSK public key
  - L is the length of the digest of the underlying hash
    algorithm
```

```
The [I-D.ietf-tls-external-psk-importer] ImportedIdentity structure
is defined as:

struct {
   opaque external_identity<1...2^16-1>;
   opaque context<0..2^16-1>;
   uint16 target_protocol;
   uint16 target_kdf;
} ImportedIdentity;

and is created using the following values:

external_identity = epskid
context = "tls13-bsk"
target_protocol = TLS1.3(0x0304)
target_kdf = HKDF_SHA256(0x0001)

The EPSK and ImportedIdentity are used in the TLS handshake as
specified in [I-D.ietf-tls-external-psk-importer].
```

TLS authentication w/DPP bootstrapping keys



TEAP w/DPP bootstrapping keys



Supplicant's subsequent connection uses provisioned certificate

Where we are and where to?

- Specification:

  draft-friel-tls-eap-dpp-05

- Running code:

  https://github.com/upros/mint

- TLS WG Reviews
  - RFC 8773 and RFC 7250 approach recommended by TLS WG at IETF 110
  - Updated draft using this recommendation reviewed at IETF 111
  - Use draft-ietf-tls-external-psk-importer suggested after IETF 113
- Rough consensus:

  adoption as a work item?