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 on hold
  - Not yet published as an IETF draft
- **To**: draft-ietf-tls-external-psk-importer
  - Specify how to convert DPP bootstrapping key to ImportedIdentity

```c
epsk = HKDF-Expand(HKDF-Extract(<>), bskey),
      "tls13-imported-bsk", L)
epskId = HKDF-Expand(HKDF-Extract(<>), bskey),
       "tls13-bpsk-identity", L)
```

The [I-D.IETF-TLS-EXTERNAL-PSK-IMPORTER] ImportedIdentity structure is defined as:

```
struct {
  opaque external_identity<1...2^16-1>;
  opaque context<6..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

Legend:
- present for dpp
- existing exchange

Client
------
ClientHello
+ cert_withExtern_psk
+ client_cert_type=RawPublicKey
+ key_share
+ pre_shared_key

Server
------
ServerHello
+ cert_withExtern_psk
+ client_cert_type=RawPublicKey
+ key_share
+ pre_shared_key

{EncryptedExtensions}
{CertificateRequest}
{Certificate}
{CertificateVerify}
{Finished}

{Certificate}
{CertificateVerify}
{Finished}

[Application Data]
TEAP w/DPP bootstrapping keys

- Authenticating Peer
  ---------------------
  EAP-Request/Identity
  EAP-Type=TEAP (TLS Start).

- Authenticator
  ---------------------
  EAP-Request/Identity
  EAP-Request/EAP-Type=TEAP (TLS Start)

Text:

authenticate TEAP with TLS-DPP using bootstrapping key

PKCS#10 TLV

PKCS#7 TLV

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?