draft-friel-tls-eap-dpp-03

Dan Harkins & Owen Friel

TLS WG, IETF 111

Changes since IETF110 / draft-02

- As agreed on mailer…
- No longer
  - Using draft-jhoyla-tls-extended-key-schedule
  - Injecting additional static-ephemeral ECDH keypair into key schedule
- Instead
  - Derive PSK from DPP bootstrap public key
  - Using RFC 8733 “TLS1.3 Cert Based Authentication with an External PSK”
  - Using RFC 7250 “Using Raw Public Keys in TLS and DTLS”
  - Server proves knowledge of bootstrap public key via PSK
  - Client proves knowledge of private key using RFC 7250 based authentication

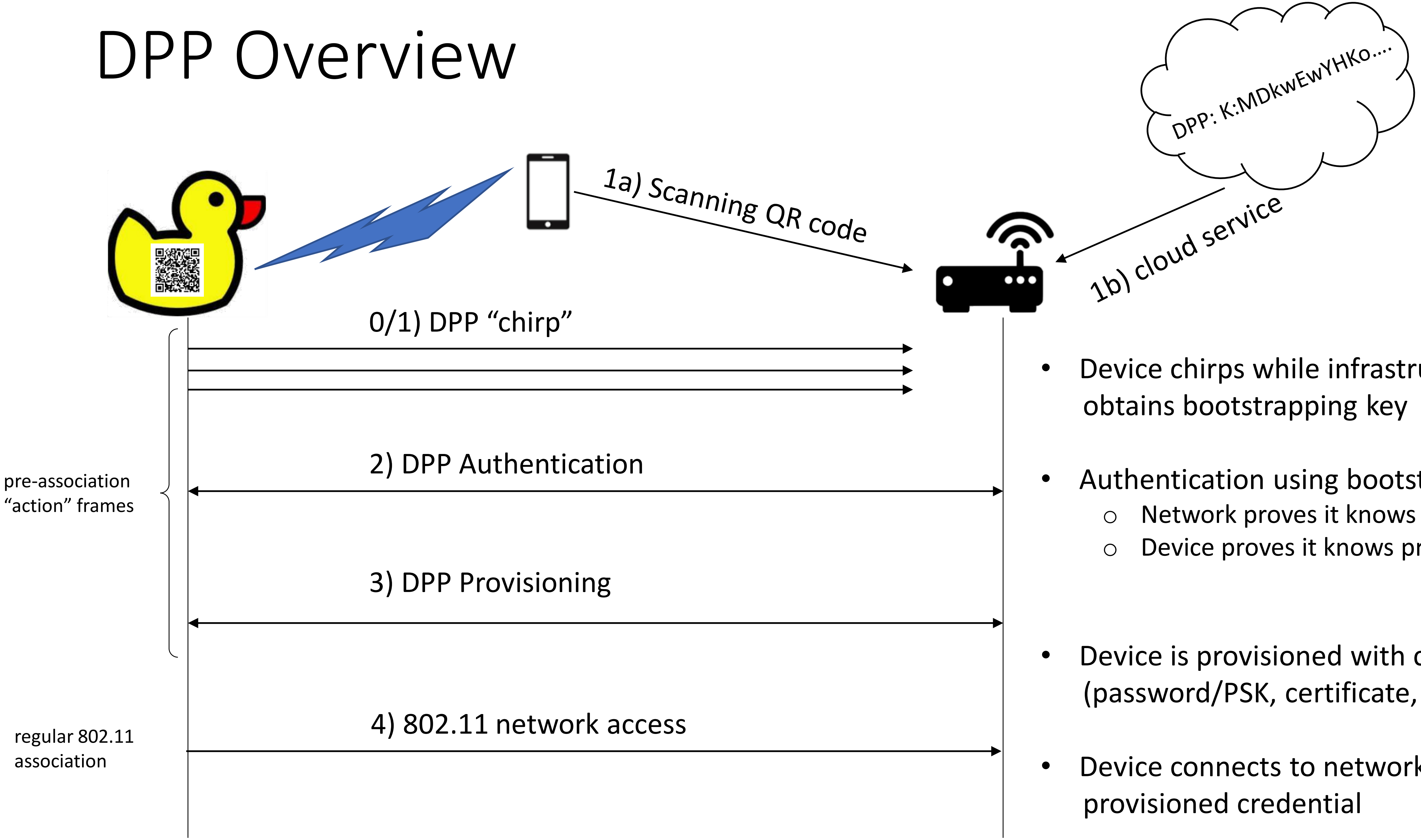
Context

- Wi-Fi alliance Device Provisioning Protocol defines how a supplicant’s bootstrap keypair can be used to authenticate the supplicant and provision it for a Wi-Fi network
- DPP and bootstrap keypair guarantee that:
    - The supplicant is connecting to a network that knows its bootstrap public key
    - The supplicant proves to the network that it knows the associated private key
- Trust model and security is based on knowledge of bootstrap key
    - The bootstrap public key is ‘secret’ and known only to the owner / network operator
    - Bootstrap ublic key is never sent in cleartext in DPP protocol
    - The private key is known only to the suppliant (e.g. embedded in the device)
- Bootstrap Public key:
    - Encoded using the ASN.1 SEQUENCE SubjectPublicKeyInfo from RFC5280

      ```
      DPP:I:GS-803XL;K:MDkwEwYHKoZIzj0CAQYIKoZIzj0DAQcDIgAC8YIhb0MFjXZzwIS3Ry9c4UAR+VZutTkYnjNLNWWGedE=;;
      ```

    - A raw keypair – does not have to be part of a PKI
    - May be static, embedded in the supplicant, and printed in a QR label, included in a BOM, etc.
    - Could be obtained from vendor cloud for true zero-touch experience
    - May be dynamically generated and displayed on a GUI
- We want to reuse the same bootstrap public key to enable a device to securely bootstrap against a wired network using EAP-TLS

DPP Overview



DPP: K:MDkwEwYHKo....

1a) Scanning QR code

1b) cloud service

0/1) DPP “chirp”

2) DPP Authentication

3) DPP Provisioning

4) 802.11 network access

pre-association
“action” frames

regular 802.11
association

- Device chirps while infrastructure obtains bootstrapping key

- Authentication using bootstrapping key
  - Network proves it knows public key
  - Device proves it knows private key

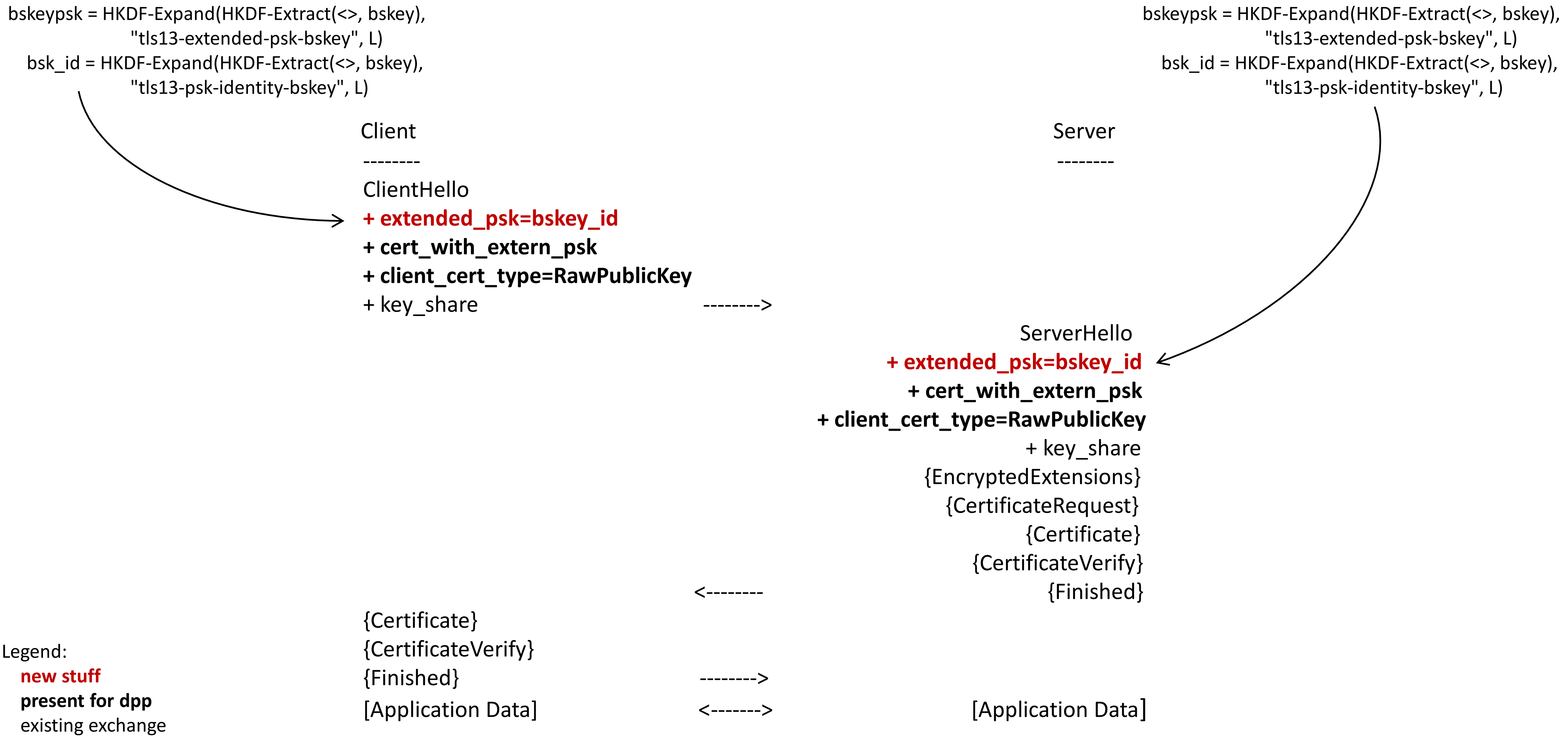
- Device is provisioned with credential (password/PSK, certificate, connector)

- Device connects to network with provisioned credential

TLS Authentication w/DPP Bootstrapping keys

- Bootstrapping key is used to derive two pieces of data
  - Identifier to signal which bootstrapping key to use for authentication
  - PSK for TLS authentication
- Use RFC 8773 “TLS1.3 Cert Based Authentication with an External PSK”
  - PSK derived from bootstrapping key is injected into key schedule
  - Client and server prove knowledge of PSK (and therefore bootstrapping public key)
- Use RFC 7250 “Using Raw Public Keys in TLS”
  - Client signs with bootstrapping private key, proves possession of private key to server
- Use draft-group-tls-extensible-psks
  - Client signals the derived PSK identity and type in extended_psk extension
- No TLS changes/extensions required over and above defining new BSK type for draft-group-tls-extensible-psks

TLS authentication w/DPP bootstrapping keys



bskeypsk = HKDF-Expand(HKDF-Extract(<>, bskey),
"tls13-extended-psk-bskey", L)
bsk_id = HKDF-Expand(HKDF-Extract(<>, bskey),
"tls13-psk-identity-bskey", L)

bskeypsk = HKDF-Expand(HKDF-Extract(<>, bskey),
"tls13-extended-psk-bskey", L)
bsk_id = HKDF-Expand(HKDF-Extract(<>, bskey),
"tls13-psk-identity-bskey", L)

Client

Server

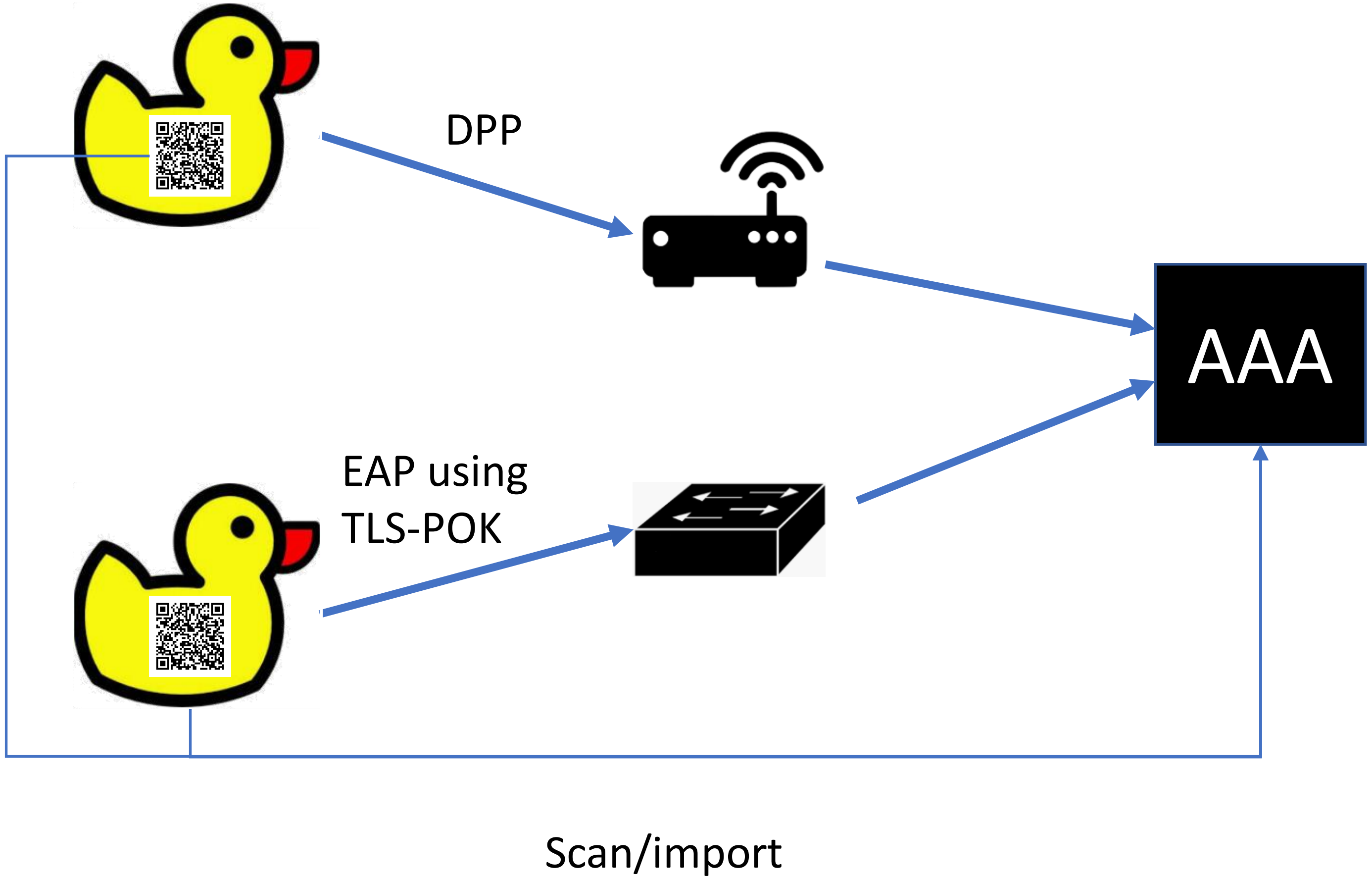
ClientHello
+ extended_psk=bskey_id
+ cert_with_extern_psk
+ client_cert_type=RawPublicKey
+ key_share --------->

ServerHello
+ extended_psk=bskey_id
+ cert_with_extern_psk
+ client_cert_type=RawPublicKey
+ key_share
{EncryptedExtensions}
{CertificateRequest}
{Certificate}
{CertificateVerify}
<--------- {Finished}

{Certificate}
{CertificateVerify}
{Finished} --------->
[Application Data] <-------> [Application Data]

Legend:
new stuff
present for dpp
existing exchange

DPP Bootstrap Key usable across Wired and Wi-Fi networks



Where we are and where to?

- Specification:

  draft-friel-tls-eap-dpp-03

- Running code:

  https://github.com/upros/mint/tree/tls-pok

- EMU

  Interest in progressing at IETF109

  Update being presented at IETF111 on Thursday