Summary

- The TLS protocol is widely deployed and used over the Internet.
  - Binary encoding, wide adoption
- There is an increasing need in the Internet to set up efficient key distribution infrastructures.
- This draft proposes a keying infrastructure based on the TLS protocol.
- Differences with the draft-ietf-tls-extractor
  - Computed keys
    - In draft-ietf-tls-extractor-01.txt the TLS PRF function is used
    - In draft-urien-tls-keygen-00.txt a separate KDF function is used
  - Pushed keys
    - The point that is not addressed by draft-ietf-tls-extractor-01.txt, is the case for which keys are pushed by server and not computed by both parties (client and server)
Full and abbreviated mode choreography

**STEP 1**
Pure TLS

**STEP 2**
Keys = KDF(master-secret, client-random, server-random, label)
Full and abbreviated mode choreography

What is available at the end of STEP1.

- Client-random, Server-random, master-secret, negotiated cipher-suite
- Implicit KDF function, with an implicit label
  - Other KDF functions MAY be negotiated via TLS-Extensions
- Cryptographic keys (Kc and Ki), used by encryption algorithms (Kc) and MAC procedures (Ki), are derived according to TLS specifications, but use KDF in place of the TLS PRF function.
  - Keys = KDF(master-secret, client-random, server-random, label)

What MAY be done during STEP2.

- Keys MAY be sent encrypted in an AVP container
- A new label (for the KDF function) MAY be sent in an AVP container.
Peer to Peer Mode

Client

Keys = KDF(master-secret, client-random, server-random, label)

Keys

Server

Keys
Distributed Mode

- Encrypted Keys
  - Client1
    - Keys are sent in AVP containers
    - Keys
  - Server
    - Encrypted Keys
  - Client2
    - Keys
AVP Container Coding (imported from TTLS)

- The AVP Code is four octets
  - Combined with the Vendor-ID field if present, identifies the attribute (i.e. the container structure) uniquely.
- The 'V' (Vendor-Specific) bit indicates whether the Vendor-ID field is present.
- The 'M' (Mandatory) bit indicates whether support of the AVP is required.
- The 'r' bits are unused and set to 0 by the sender.
- The AVP Length field is three octets, and indicates the length of this AVP including the AVP Code, AVP Length, AVP Flags, Vendor-ID (if present) and Data.
- Data Length is two octets and indicates the size of data without padding bytes.

```
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| AVP Code                      | AVP Length                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| V M r r r r r r               | Vendor-ID (opt)               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Data Length                   | Data                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Optional padding bytes        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
KDF Consideration

Based on the work

Krawczyk, H, "On Extract-then-Expand Key Derivation Functions and an HMAC-based KDF", http://www.ee.technion.ac.il/~hugo/kdf/, March 2008

Already use in IKEV2

For Transform Type 2 (Pseudo-random Function), defined Transform IDs are:

Keying material is generated according to the selected prf

\[ \text{prf}^+ (K, S) = T1 \mid T2 \mid T3 \mid \ldots, \text{where} \]

\[ T1 = \text{prf} (K, S \mid 0x01), T2 = \text{prf} (K, T1 \mid S \mid 0x02), T3 = \text{prf} (K, T2 \mid S \mid 0x03) \]

\[ \text{SKEYSEED} = \text{prf} (\text{Ni} \mid \text{Nr}, g^{\text{ir}}), \quad \text{SK}_d = \text{prf}^+ (\text{SKEYSEED}, \text{Ni} \mid \text{Nr} \mid \text{SPIi} \mid \text{SPIr}) \]

\[ \text{KEYMAT} = \text{prf}^+ (\text{SK}_d, \text{Ni} \mid \text{Nr}) \text{ or } \text{KEYMAT} = \text{prf}^+ (\text{SK}_d, g^{\text{ir}} \mid \text{Ni} \mid \text{Nr}) \]
Proposed default KDF

Notations.
- The first argument to a keyed function denotes the key, the value $K$ is the key to PRF and $x$ its input.
- The symbol $\|\|$ denotes concatenation.
- Given two numbers $N$ and $n$ the symbol $N:n$ represents the value $N$ written as $n$-bit integer.
- $L$ is the length in bits, of this output value delivers by the HMAC procedure ($L=160$ for SHA1)

Pseudo Random Key
- $PRK = HMAC$ (client-random $\|\|$ server-random, master-secret)

Keying material, whose length in bits is $D$, required $D/L$ operations.
- First is expressed as
  - $K(1) = HMAC(PRK, 0:L \| KeyLabel \| 0:32)$,
- Further operations (whose number is $i$) are computed according to
  - $K(i+1) = HMAC(PRK, K(i) \| KeyLabel \| i:32)$
  - where $KeyLabel$ is an ASCII string set to "key expansion".
Questions ?