Use of Variable-Length Output PRFs in IKEv2

draft-smyslov-ipsecme-ikev2-prf-plus-00

Valery Smyslov

svan@elvis.ru

IETF 122

Use of PRF in IKEv2

- Pseudorandom Function (PRF) is used in IKEv2 for authentication and key derivation
- PRF is assumed to be a function that takes a variable-size key and variable-size string and produces a **fixed-size** pseudorandom output; it is denoted as *prf* in RFC 7296
- For longer output a construction *prf+* is used:

```
prf+(K,S) = T1 | T2 | T3 | T4 | ...
T1 = prf(K, S | 0x01)
T2 = prf(K, T1 | S | 0x02)
T3 = prf(K, T2 | S | 0x03)
T4 = prf(K, T3 | S | 0x04)
```

Concrete PRFs in IKEv2

- Concrete PRF is negotiated via `PRF` transform in `SA` payload
  - *prf*+ construction is not negotiated, it is fixed in IKEv2 specification
- Currently all registered PRFs for IKEv2 have fixed output length
- Recently new PRFs appeared that can produce as many pseudo-random bits as requested in one call
  - KMAC as an example
  - more such PRFs can appear in future

Use of PRFs with Variable-Length Output in IKEv2

- These PRFs can be used as fixed-length output PRFs (by fixing the output length)
  - suitable for use as *prf*
  - inefficient for use as *prf*+
- Optimization: for *prf*+ use cases these PRFs can be used with only one iteration:

  ```
  prf+(K,S) = prf(K, S | 0x01)
  ```

  - proposed in draft-salter-ipsecme-sha3 for KMAC
  - requires special handling of these PRFs in IKEv2
  - *prf*+ construction becomes degenerate

Can We do Better?

- If PRFs with variable-length output have to be handled specially in IKEv2, then why not get rid of *prf*+ for them?

```
prf+(K,S) = prf(K,S)
```

  - – code complexity is the same as for using single iteration of *prf*+
  - – no need for additional fixed input byte 0x01 (former counter)
  - – also discussed as a possibility in draft-salter-ipsecme-sha3

Generic Rules?

- Use of variable-length output PRFs should be specified as in a generic way, not for each such PRF
  - for each such PRF a preferred key size should be specified
  - in case of *prf,* use these PRFs with output length equal to the preferred key size
  - do **not** use *prf+,* instead do a single call to these PRFs with a needed output length
  - do not use customization strings if they can be set by API (since APIs may vary)

Thanks!

Comments?

WG adoption?