Babel Information Model

Barbara Stark

Mahesh Jethanandani

IETF 110, March 2021

Still need to resolve 1 DISCUSS...

... and we’re not doing anything else until we do!

> **babel-mac-key-value: ... This value is of a length suitable for the associated babel-mac-key-algorithm. If the algorithm is based on the HMAC construction [RFC2104], the length MUST be between 0 and the block size of the underlying hash inclusive (where "HMAC-SHA256" block size is 64 bytes as described in [RFC4868]). If the algorithm is "BLAKE2s-128", the length MUST be between 0 and 32 bytes inclusive, as described in [RFC7693].**

We went around on this twice, and this is where we landed after the 2nd time.

Some of what we were trying to avoid running into:

OSPF (RFC 5709) and RFC 2104 differ on how to calculate a cryptographic key from a long authentication key. They differ in what they consider “long”.

What RFC 5709 describes is different than what RFC 2104 (HMAC) describes. RFC 2104 says to create a hash of the authentication key (K) if it is longer than the block size of the hashing algorithm (B).
For SHA-256, B != L. Therefore, using RFC 5709 will get a different value for the cryptographic key than RFC 2104, for authentication key length between L and B. This difference was noted in RFC5709 errata.
RFC 7166 updated RFC 6506 wrt this. But RFC 7166 didn't update RFC 5709. RFC 7474 updated RFC 5709 and kept the L boundary.

and converting ASCII input.

But the current text may not work for the OSPF v RFC2104 problem

Representing as binary is ok.

But Ben K points out that the current requirement does allow hitting the code that would result in inconsistent key values, since the stated limit for HMAC-SHA256 is 64 bytes, not 32.

So what do we need to provide a Babel MAC implementation as input?

Closed Issues

I asked the group about router-id restrictions and configuration of interval values. The conclusion was:

- No info model restrictions on router id.
- No configuration of interval values.