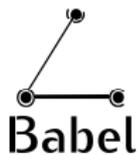


# About some Babel drafts

28 March 2017



# Drafts

I am currently editor of:

2 base drafts in scope:

- rfc6126bis (**urgent**);
- applicability statement (**urgent**).

3 extension drafts (out of scope):

- source-specific routing (**urgent, not ready**);
- rtt-based routing (not urgent, **ready**);
- diversity-based routing (not urgent, not ready).

# rfc6126bis

Rfc6126bis is the **merger** of RFCs 6126 and 7557, plus:

- **bug fixes**;
- **clarifications**;
- weakly **compatible changes**  
(break the letter of the spec, but not existing implementations).

## rfc6126bis: about RFC 6126

RFC 6126 was published in 2011. Since then:

- four important extensions;
- 3 independent reimplementations of Babel;
- a few bugs and minor omissions in the spec (but good enough for independent implementation).

## rfc6126bis: about RFC 7557

**RFC 7557** (extension mechanism) was published in 2015, **after** a number of extensions had been designed and deployed.

- RFC 6126 **reserves space** for extensions, but doesn't define their format;
- RFC 7557 written **after extensions were designed**, implemented and deployed:
  - defines the **format of sub-TLVs**;
  - does not define the format of the packet trailer (never used).

Two distinct RFCs for **purely historical reasons**.

Rfc6126bis intends to **merge RFC 6126 and RFC 7557**.

# Status of rfc6126bis

Status of rfc6126bis:

- bug fixes: **done**;
- clarifications needed:
  - neighbour acquisition;
  - sending of requests;
- merger of RFC 7557: **in progress**;
- weakly-compatible changes:
  - unicast Hellos;
  - redefine updates.

## Neighbour acquisition in rfc6126bis

Very technical **discussion** on the list. **2 opinions:**

- **majority opinion:** **leave it vague**  
MUST eventually acquire any neighbour it wishes to exchange routes with;  
provide implementation suggestions;
- **minority opinion:** **specify precisely**  
using a finite-state automaton;  
solves a problem with HMAC-based security (RFC 7298).

**Leave it vague.**

**(Allow extensions to tighten the rules?)**

Consensus?

## rfc6126bis: sending requests

**Sending requests** is the most tricky bit of Babel.

The description in RFC 6126 is **a mess** (badly written), but it turns out to be **good enough** for independent reimplementation.

**Rewrite.**

Make it **slightly more permissive**.

See my mail to the list dated 6 December 2016.

# Unicast Hellos

All Babel TLVs can be sent over **unicast or multicast**, and have the **same meaning**.

With the **exception of Hellos**.

Just **sending Hellos over unicast doesn't work**: per-interface seqno counter.

At least two active implementers are clamouring for unicast Hellos, but there is no complete design.

**Wasted opportunity?**

# Integration of the extension mechanism

For historical reasons, RFCs 6126 and 7557 are separate documents. Rfc6126bis aims to **integrate the two**.

Tricky to do right, 3 attempts so far:

1. by Toke Højland, **not true integration**, included as a separate section;
2. by me, **didn't work out**, thrown out;
3. by me, **seems to work**, not public yet.

Stylistically, a lot of drama for nothing: “here’s where you put sub-TLVs, oh, by the way, we don’t define any in this document”.

# Digression: source-specific routing

Source-specific routing is:

- a very exciting extension;
- required by Homenet.

The packet format is a mess:

- 3 new kinds of TLV;
- some have way too many fields.

# Digression: source-specific routing

Source-specific routing is:

- a very exciting extension;
- required by Homenet.

The packet format is a mess:

- 3 new kinds of TLV;
- some have way too many fields.

Solutions:

- add a mandatory bit to sub-TLVs — breaks compatibility
- use the AE mechanism!

# Changes to the AE mechanism

Updates and requests have an **AE field** (1 octet).

**Address Encoding**, determines the interpretation of the payload of an update (IPv4, IPv6, etc.).

Idea: use the **normal update and request TLVs** for source-specific updates, but with **a new AE value**.

## Changes to the AE mechanism (2)

Use the update TLV for source-specific updates, but with a new AE value.

Needs [changes to the base spec](#):

- making the format of updates **less rigid**;
- defining how **compression** works with unknown extensions.

Two [competing approaches](#) and a half:

- make the [payload](#) of updates **opaque** (current favourite);
- make [updates as tightly specified](#) as possible;
- forget it, use a **mandatory bit** on sub-TLVs.

We need [more examples](#):

- Gwendoline Chouasne: ToS routing in Babel;
- BIER?

# Applicability statement

The [Babel Applicability Statement](#) has a [long history](#).

1. “It’s a routing protocol, it routes”  
(too short);
2. draft-chroboczek-babel-doesnt-care-00  
(too funny);
3. draft-ietf-babel-applicability-statement-01  
(too sober).

## Applicability statement — first try

“It’s a routing protocol, it routes”.

While technically correct, this was considered too short and not informative enough.

## Applicability statement — second try

[draft-chroboczek-babel-doesnt-care-00](#)

“The best IETF draft ever.” — DT

“Reminds me of the Honey Badger” — TL

“It’s **not an applicability statement**, it’s a (screamingly funny) **piece of bragging**” — Anonymous

## Applicability statement — third try

[draft-ietf-babel-applicability-statement-01](#)

Very sober document: **no bragging** (I swear!), only describes **existing deployments**.

# Applicability statement — third try

## draft-ietf-babel-applicability-statement-01

Very sober document: **no bragging** (I swear!), only describes **existing deployments**.

Reviewed by Alexander Vainshtein (thanks!):

1. **needs introduction**;
2. needs **more precise data** about existing deployments;
3. needs description of used **extensions**.

I strongly agree with (1) and (3).

I don't disagree with (2).

# Conclusion

We aim for:

- complete draft of rfc6126bis in Prague  
main stumbling blocks:
  - unicast Hello;
  - redefinition of updates (AE);
- complete draft of applicability statement in Prague  
main stumbling block: it's boring.
- new draft (out of scope) for source-specific routing;
- first draft of ToS routing?