

CoAP Protocol Indication

`draft-amsuess-core-transport-indication-03`

Christian Amsüss

2022-03-25

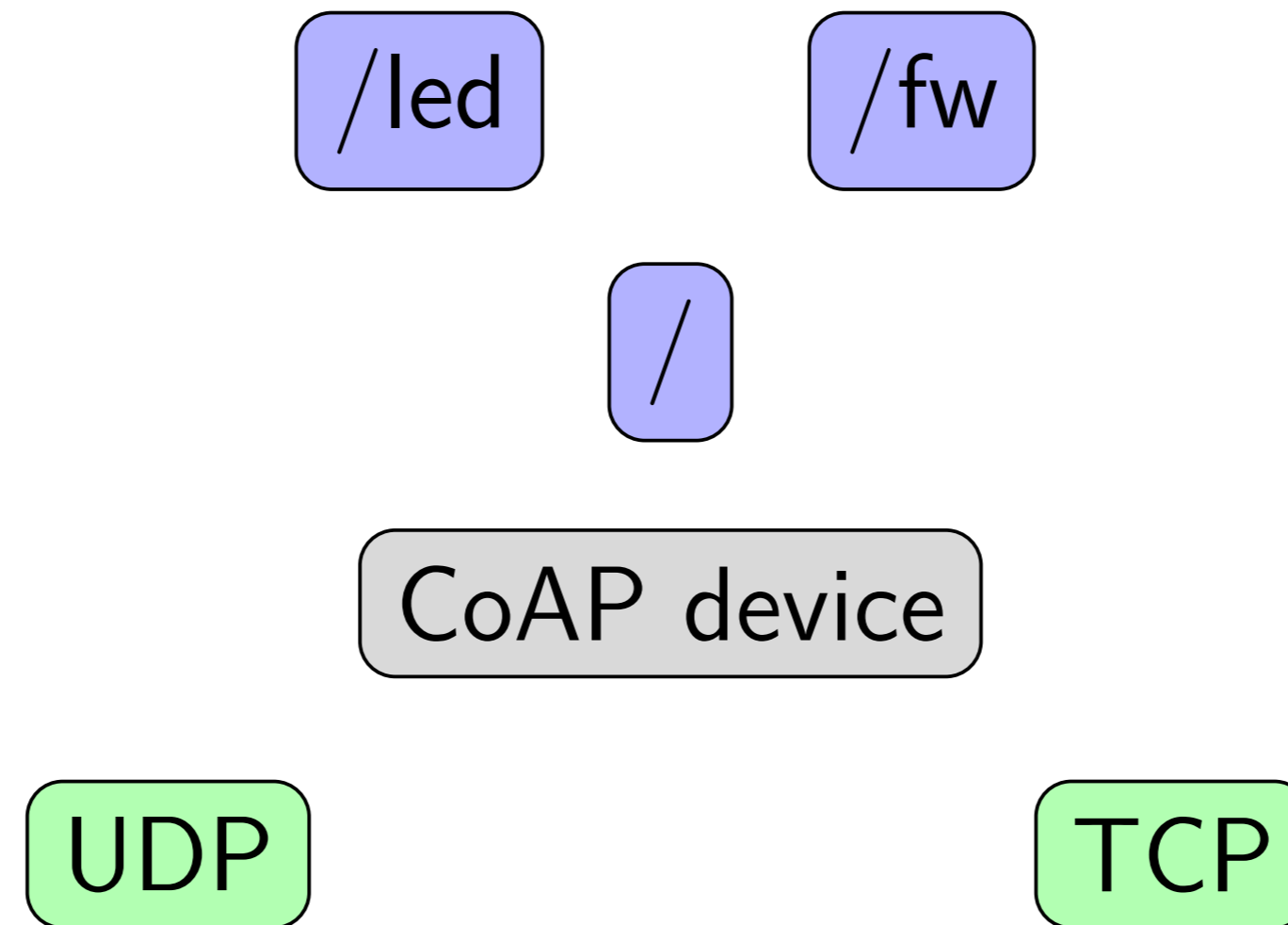
CoRE at IETF 113 in Vienna

The badge addon that sums it up

Big Thank You to the secretariat for making meetings even more colorful

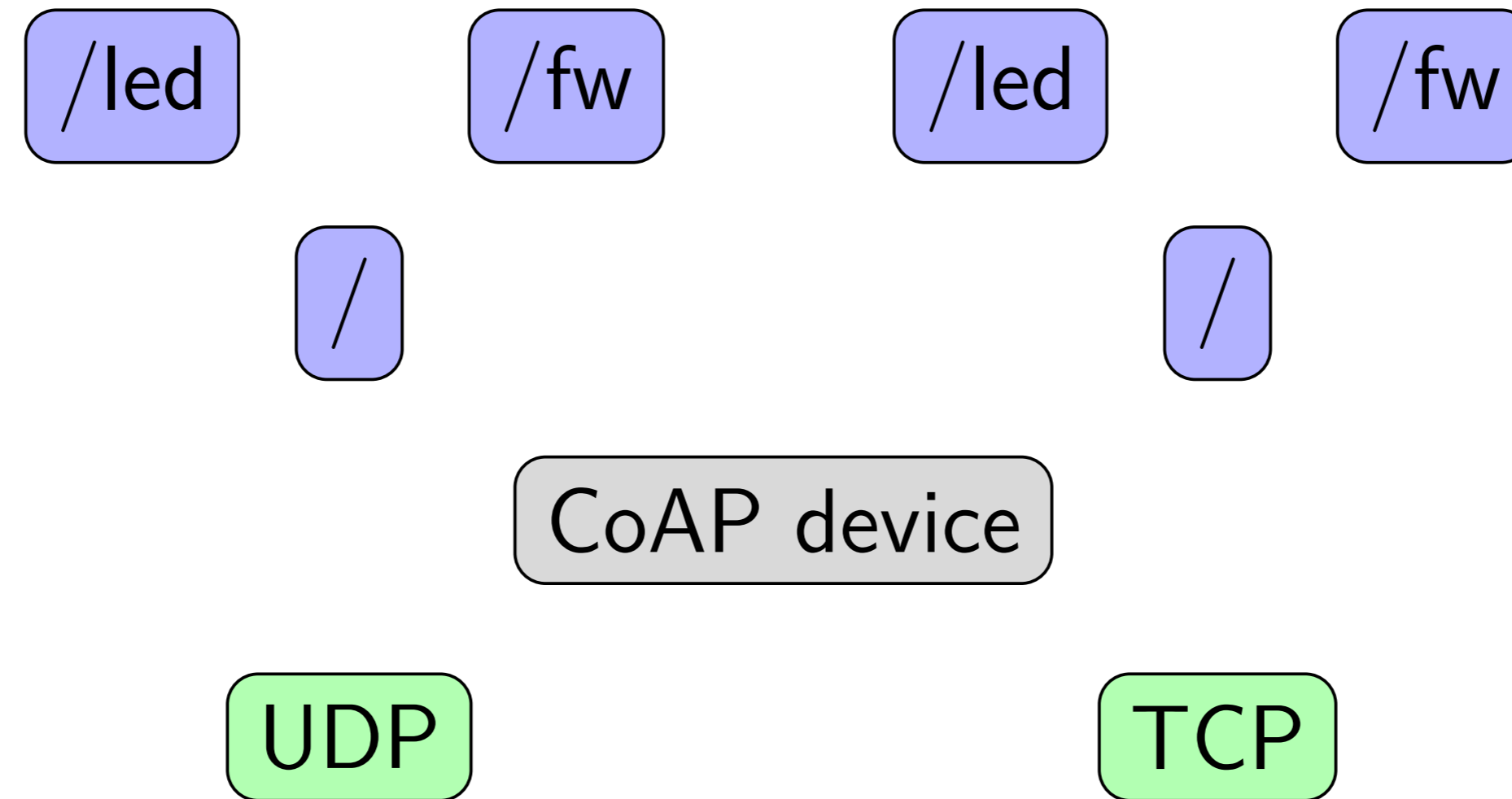


Problem description



Not just TCP/UDP: (D)TLS and WebSockets are specified; there are drafts for SMS, serial lines and Bluetooth, and ideas for QUIC via TAPS.

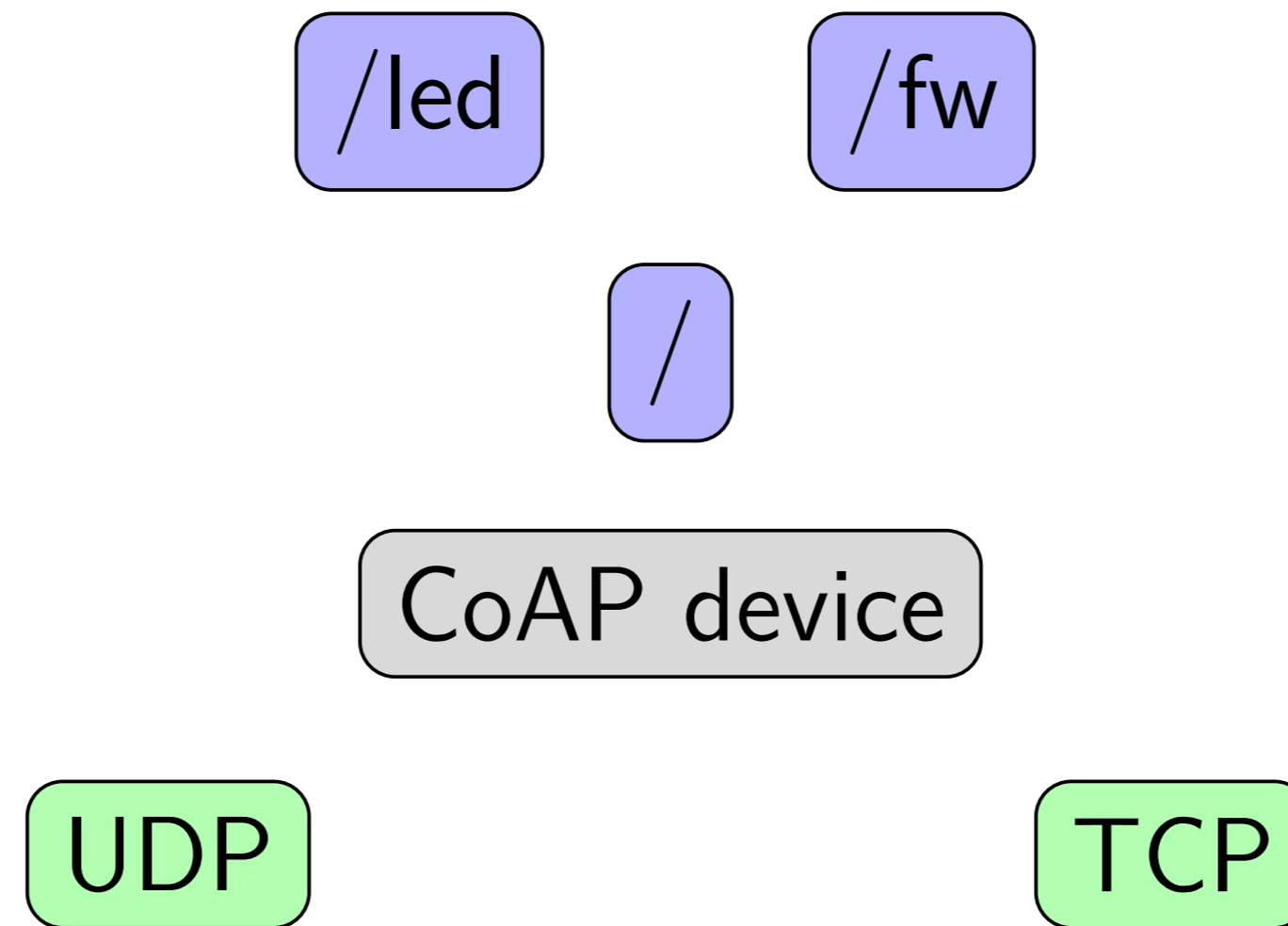
Problem description



`coap+tcp://[2001:db8::1]/led` \neq `coap://[2002:db8::1]/led`

Server MAY do this, client MUST NOT assume it. See `http` vs. `https`.

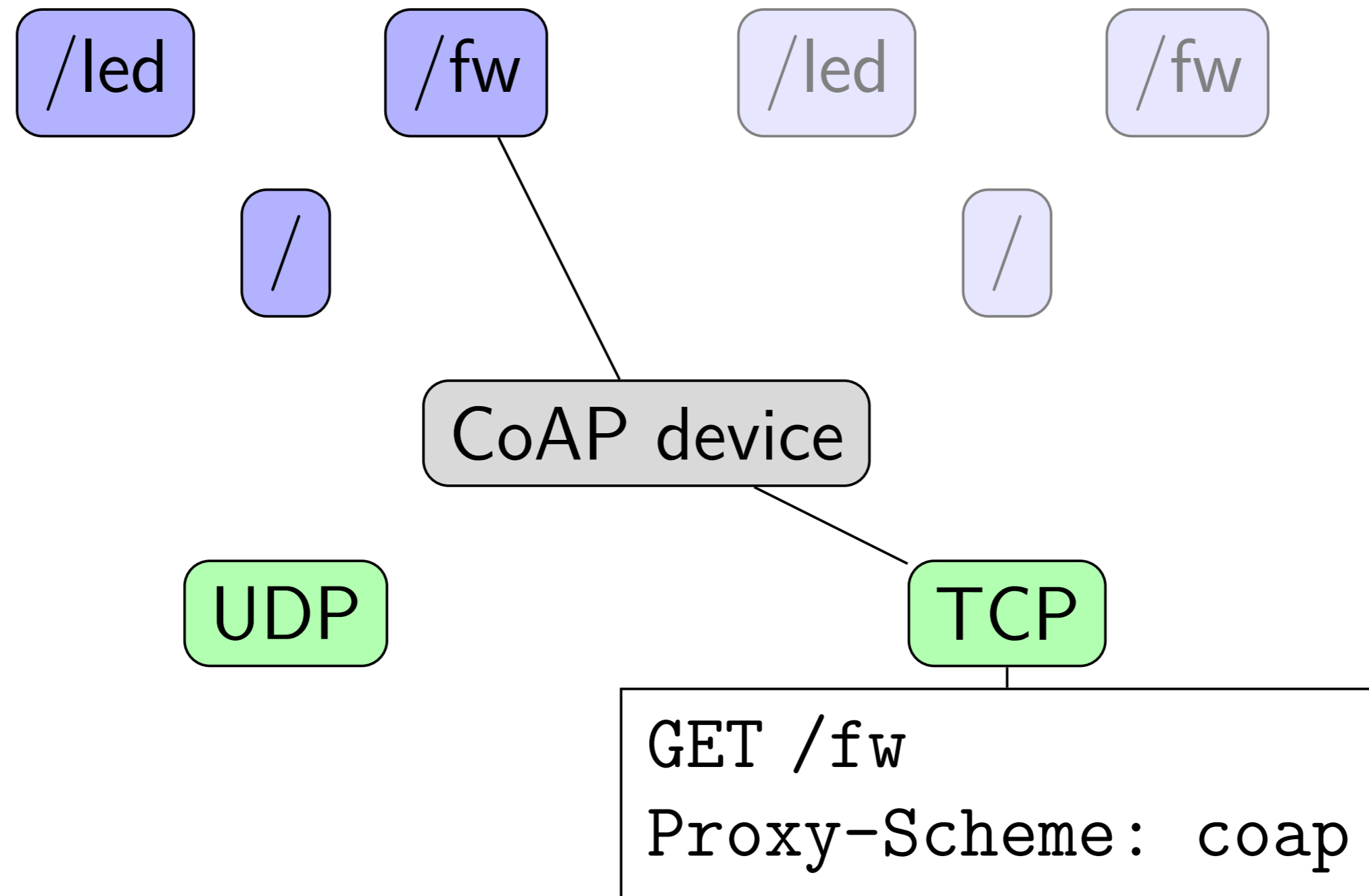
Problem description



`coap+tcp://[2001:db8::1]/led` $\stackrel{?}{:=}$ `coap://[2002:db8::1]/led`

- “aliasing” – generally discouraged on URIs.
- Multiple cache entries.
- What does identity mean for, say, block-wise operation?
- What does identity mean security?
- Breaking change.

Proposed mechanism: We have the runtime parts

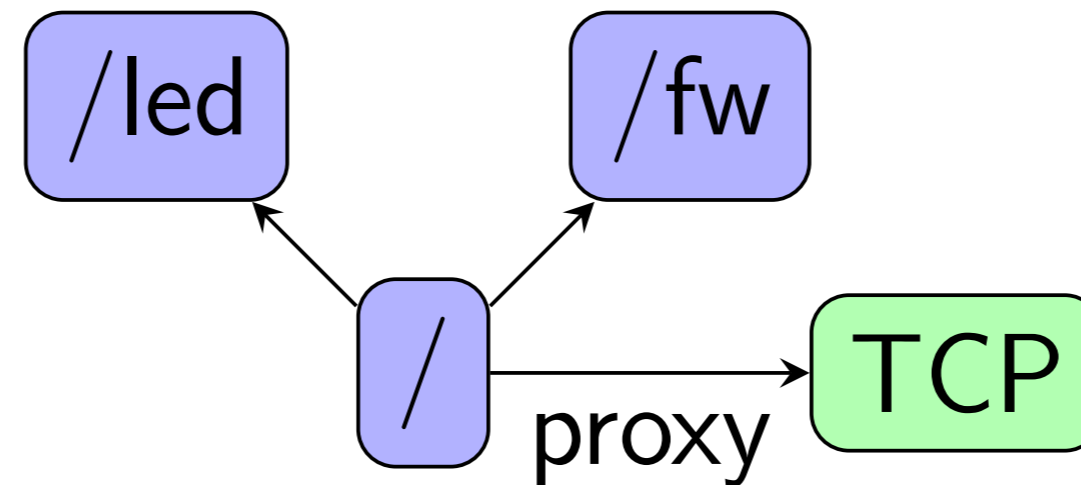


The option is already there, and has the right meaning:

RFC 7252 Section 6.5 gives `coap://[2001:db8::1]/fw` – the intended URI.

“Being a proxy” is a big word – it’s really just processing that option.

Proposed mechanism: We need discovery



“the proxy is” `TCP to 2001:db8::1` \mapsto “proxy identified as” `coap+tcp://[2001:db8::1]`

```
</fw>;rt="tag:...:suit";rel=hosts;anchor="/",  
<coap+tcp://[2001:db8::1]/>;rel=has-proxy;anchor="/"
```

Goals (1-2/5)

Enablement Inform clients of the availability of other transports of servers.

No Aliasing Any URI aliasing must be opt-in by the server. Any defined mechanisms must allow applications to keep working on the canonical URIs given by the server.

Just As With Any Proxy.

OK, there's more in the text, but that's the gist.

- Any requirements on connecting directly apply also on the connection through a proxy.
- Proxies that do not have the relevant precise credentials are out of scope.
- Users worrying about traffic misdirection can decide to only use a proxy statement if it comes from an authoritative source.

Is this enough?

Is this enough?

Maybe.

Is this enough?

Maybe.

Proxy may be provided by anyone (not ruled out so far, but not described either).

And we still send 5 bytes per request...

Proxy interaction

Goals (4-5/5)

Proxy usability All information provided must be usable by aware proxies to reduce the need for duplicate cache entries.

Proxy announcement Allow third parties to announce that they provide alternative transports to a host.

Proxies see a single resource. Proxies may use it to pick their upstream.

External components (e. g. a Resource Directory) can state that they provide proxying services.

Whether a client uses them depends on client's security requirements; as a minimum, application and transport layer security must not deteriorate. (Generally trivial with OSCORE and out of scope for (D)TLS).

Eliminating per-request overhead

Goals (3/5)

Optimization Do not incur per-request overhead from switching protocols. This may depend on the server's willingness to create aliased URIs.

```
<coap+tcp://[2001:db8::1]/>;rel=has-unique-proxy;anchor="/"
```

Request can go to `coap+tcp://` on the wire; application still think in terms of `coap://`. Looks like aliasing, but that's a matter of perspective (SCHC?).

Also answers: "Do I have to send the Uri-Host option?"

Eliminating per-request overhead: Security

Open issue: Potential confusion about intended resource.

Possible solution: Accept unique-proxy statement only when coming from authoritative resource.

(No big loss: Third party proxies are generally non-unique anyway).

Take-home message

- It can probably be just this simple.
- No URI aliasing introduced in applications.

Questions? Comments? Way forward?

Backup slide / FAQ

Didn't we want to do this with DNS?

We¹ still can, just need to phrase the equivalent statements in DNS.

Straw man for “coap://device.example.com has CoAP-over-TCP running on port 1234”:

```
_has-coap-proxy._tcp.device.example.com SRV 0 0 device.example.com 1234  
device.example.com AAAA 2001:db8::1
```

How does this relate to HTTP's Alt-Svc?

Generally similar; links instead of headers (as common in CoAP), and no need for protocol-id because we have schemes already.

How does this (esp. security for unique proxies) relate to CoRAL?

Yes.

¹Whoever wants to use it will need to volunteer as coauthor.