Universal Configuration Information Option

draft-troan-6man-universal-ra-option-05

Tim Winters <tim@qacafe.com>
Ole Trøan <ot@cisco.com>

IETF 111, 6man WG

https://github.com/ietf-6man/universal-ra

Changes since IETF 109 (work started IETF 104)

- Added section on CBOR encoding with a new integer to string key mapping table (to save space) as suggested by Thaler

IETF109:

- s/Experimental/Proposed Standard/g
- Added DHCPv6 support
- New co-author: Tim Winters

Problem

- Working group spends an inordinate amount of time arguing over proposed new RA (or DHCPv6) options. Some arguments go "since I don't need it, let's not standardize it.
- Does the working group add value to the set of problems where an RA is used as a general carrier?
- Every new option requires implementation changes both in router OS / management system and in host's RA processing engine

Contributions

- **Technical**: A self-describing option format. JSON objects modelled in CDDL, encoded in CBOR.

  Allows new options to be added *without* implementation changes in router OS or kernel.

- **Process**: This option space is not a constrained resource. Options can be specified directly in IANA registry with expert review.
  No 6MAN WG involvement required for defining new objects. This is how DHC options are handled now.

Universal RA/DHCPv6 format

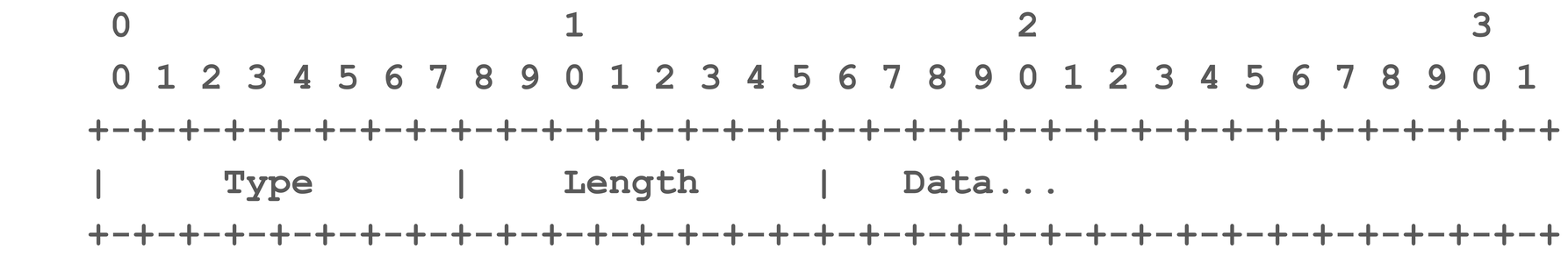

Figure 1: Universal RA Option Format

Fields:

Type 42 for Universal RA Option

Length The length of the option (including the type and length fields) in units of 8 octets.

Data CBOR encoded JSON padded to the nearest 8 octet boundary.

Can be distributed across multiple packets. Single option size is limited to 2^8*8 = 2048-2 bytes.

Only allowed in RA.

Obsolete RFC4833? ;-)

JSON object:

```
{
  "ietf": {
    "dns": {
      "dnssl": ["example.com"],
      "rdnss": ["2001:db8::1", "2001:db8::2"]
    },
    "nat64": {
      "prefix": "64:ff9b::/96"
    }
  }
}
```

```
 ietf = {
   ? dns : dns
   ? nat64: nat64
   ? ipv6-only: bool
   ? pvd : pvd
   ? mtu : uint .size 4
   ? rio : rio
}
pio = {
  prefix : tstr
   ? preferred-lifetime : uint
   ? valid-lifetime : uint
   ? a-flag : bool
   ? l-flag : bool
}
rio_route = {
  prefix : tstr
   ? preference : (0..3)
   ? lifetime : uint
   ? mtu : uint .size 4
}
```

IANA

- New IANA registry for the universal CI option.
- CDDL described objects
- Self contained in IANA registry (or a stable reference)
- Expert review
  - Expert should have the option to punt to WG if IETF document required.

Integer Keys

Changes and additions to the registry follow the policies below [RFC8126]:

Range	Registration Procedure
-23-23	Standards Action
24-32767	Specification Required
32768-18446744073709551615	Expert Review

Table 1

CDDL/JSON Mapping Parameters to CBOR

Parameter Name / JSON key	CBOR Key
ietf	-23
pio	-22
mtu	-21
rio	-20
dns	-19
nat64	-18
ipv6-only	-17

Integer keys vs string keys

```
#!/usr/bin/env python3
import cbor2
import ipaddress
print(len(cbor2.dumps({'ietf': {'pio': {'prefix':
                                        ipaddress.IPv6Network('2001:db8::/32')}}})))
print(len(cbor2.dumps({-23: {-22: {-15: ipaddress.IPv6Network('2001:db8::/32')}}})))
```

42 vs 29 bytes. (RFC4861 PIO is 32 bytes)

ADOPT OR ABANDON?