YANG Data Model for DHCPv6 Configuration

draft-ietf-dhc-dhcppv6-yang-08
Y. Cui, L. Sun, I. Farrer, S. Zechlin, Z. He, M. Nowikowski

IETF106 Singapore
What’s Happened since -08 (presented at IETF104)?

• New author joined (Michal Nowikowski)
• -09 and -10 updates posted
• -10 is a major update
  • Reduced scope
  • New structure for each of the basic element modules (client, relay, server)
  • Revised and re-worked the option definitions and their integration with the element modules
  • Implementation specific functions moved out of server module
  • A lot of clean ups and improvements in consistency across the different modules
Reduced Scope

• Trying to model the protocol in it’s current state, with all of the published extensions wasn’t working

• As proposed at IETF104, scope has been reduced to only cover RFC8415

• Modules are structured to be extensible by future work as necessary

• The resulting modules are much easier to work with – draft is now 20 pages shorter (only 74!)
Element Module Main Changes – Client & Relay

• Configuration has been integrated with the ’ietf-interfaces’ structure e.g.:

```plaintext
+-rw dhcpv6-client
  +-rw client-if* [if-name]
    +-rw if-name if:interface-ref
    +-rw type-code? int16
    +-rw (duid-type)?
    |   +-:(duid-llt)
```

• State data for lease timers now modelled
• Prefix delegation is now enabled as a feature
• Notifications reworked and reduced (some of the odd ones didn’t make sense)
Element Module Main Changes – Server

- Previously, the server module included nodes for configuring interfaces, backend database etc.

- As these are implementation specific, they have now been moved from the main element module to the appendix as an example:

```plaintext
module: example-dhcpv6-server-config
    augment /dhcpv6-server:/dhcpv6-server:/dhcpv6-server:/vendor-config:
        ++-rw serv-attributes
          ...
        +-rw lease-storage
          +-rw (storage-type)?
            ---:(memfile)
                |- ++-rw memfile-name?       string
                |- ++-rw memfile-lfc-interval? uint64
            ---:(mysql)
                |- ++-rw mysql-name?         string
                |- ++-rw mysql-host?         string
```
Client Class Selection Nodes

- Used by the server for identifying and classifying incoming client messages
- Research into how these are configured for different implementations shows that no two are alike
- So, this function has also been moved to an example module, augmented into the server element module
Modelling DHCP Options - 1

A module contains definitions for options taken from RFC8415

Identities are used to augment the option definitions into the relevant element module

This allows for simple re-usability and extensibility
To extend the option definitions, the same method is used:
- Define the options in a standalone module
- Augment the option definition into the relevant DHCP element module
- Elements which implement the option need load the module
- RFC3319 (SIP Server) Options are provided as an example in the appendix (see above)
- Guidelines for writing YANG modules for new options
Where next?

• The modules should now be ‘feature complete’ with RFC8415
• Due to the number of changes, there needs to be cleanups throughout
• Please review and comment
• Then, hopefully on to WGLC