YANG Data Model for DHCPv6 Configuration

draft-cui-dhc-dhcpv6-yang-03

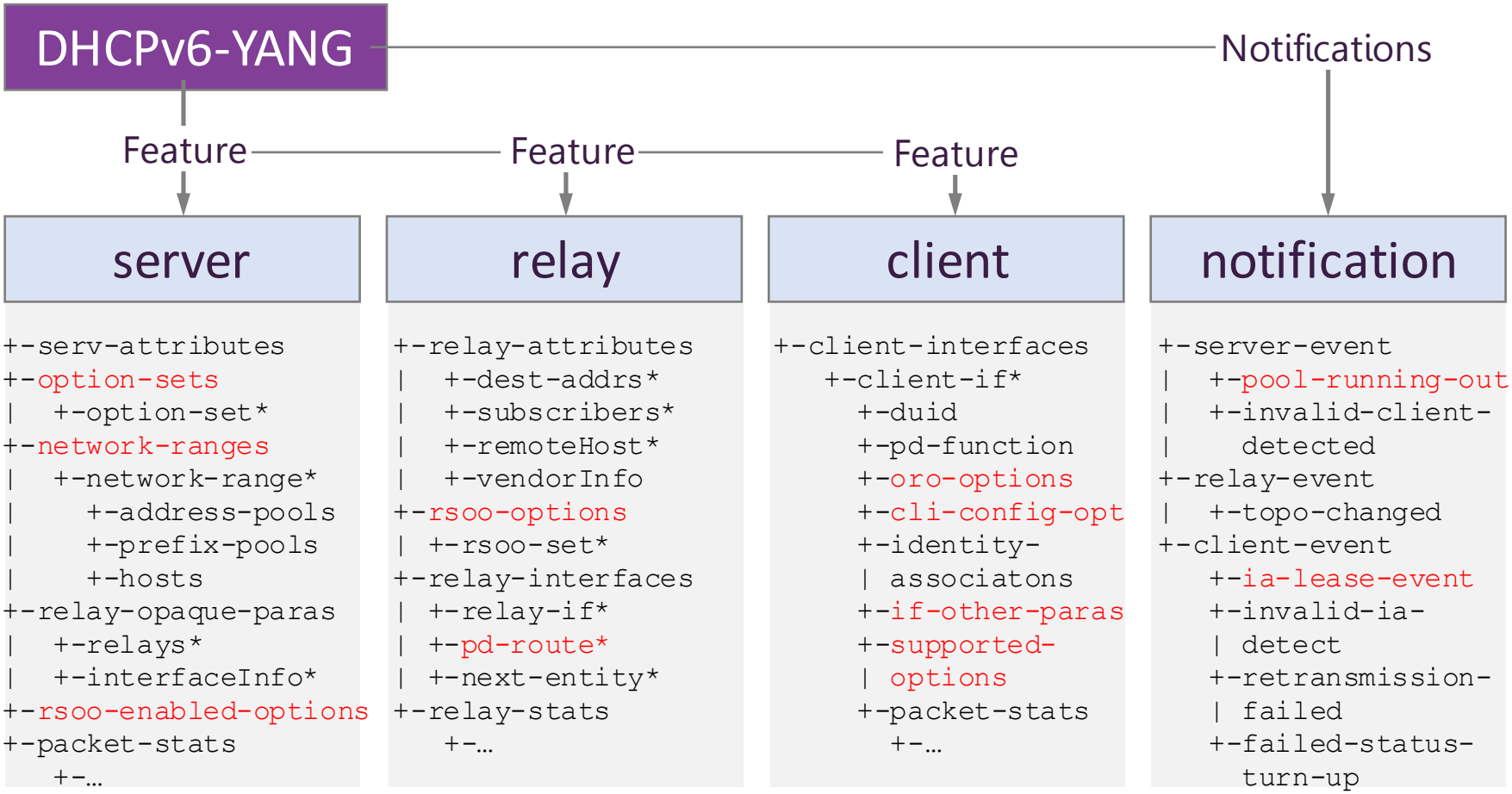
Yong Cui, Hao Wang, Linhui Sun,
Ted Lemon & Ian Farrer

IETF93 Prague

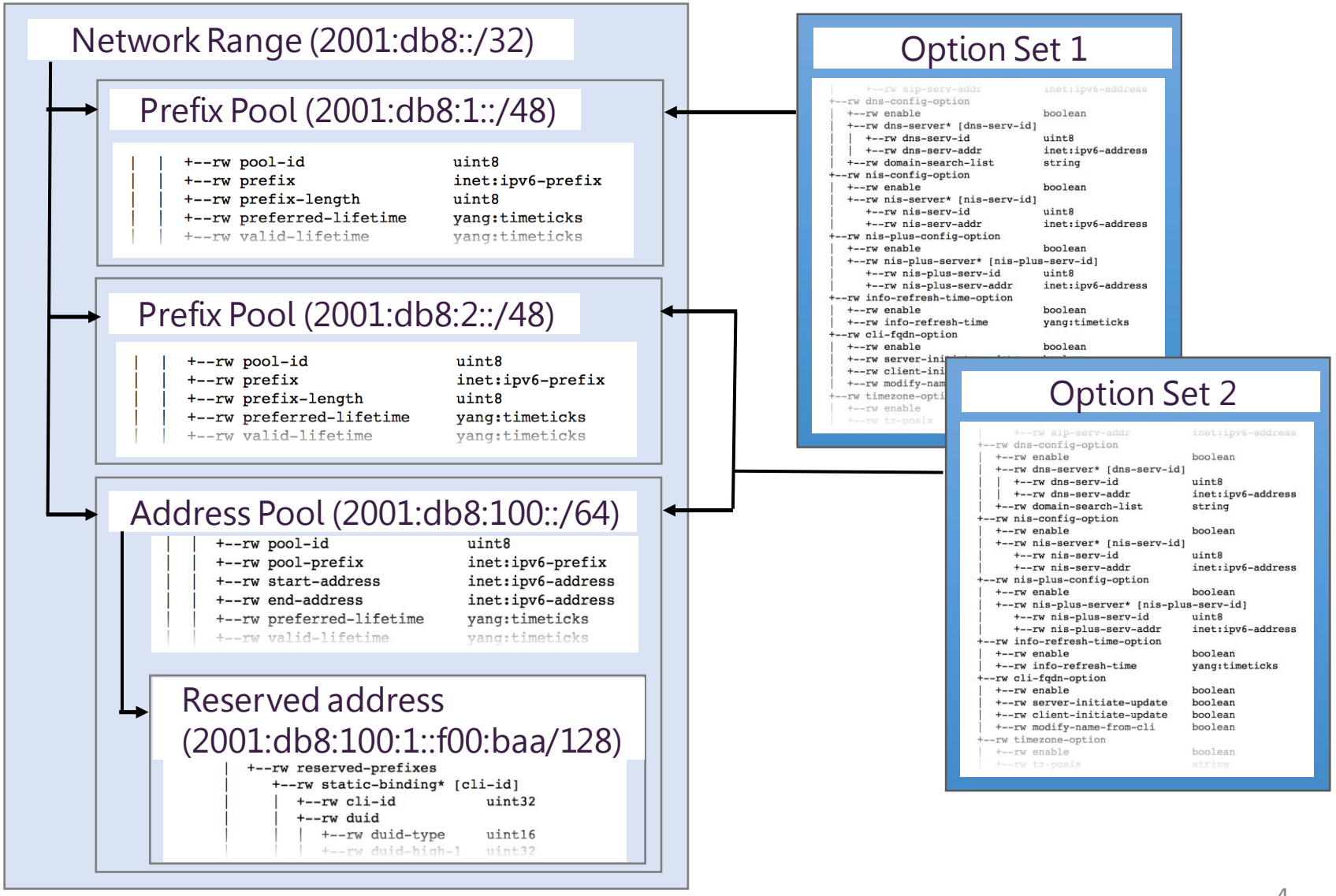
Major Changes since IETF92

- A **hierarchy** is introduced in the server portion
  - allow administrator configuring the server at different levels (network, prefix/address pool/client)
- Introduced '**option sets**' container
  - Holds all defined DHCPv6 options and allows them to be applied in the hierarchy above
- Add a paragraph in Sec.1 to
  - explain **why the client model is contained**
  - Intended for the configuration of client function and for obtaining state data (e.g. IA lease) of client which is learnt via DHCPv6
- Modify and improve the Notification module
  - Notify before things happen

Model Structure



Abstract Server Model Structure



DHCPv6 Server Option Sets

```
+--rw option-sets
     |  +--rw option-set* [option-set-id]
     |     +--rw option-set-id                  uint8
     |     +--rw user-class-value?              String
     |     +--rw enterprise-number?             Uint32
     |     +--rw store-client-link-layer?       Boolean
     |     +--rw preference-option
     |     |  +--rw enable                      boolean
     |     |  +--rw preference-value            uint8
     |     +--rw sip-server-option
     |     |  +--rw enable                      boolean
     |     |  +--rw sip-server* [sip-serv-id]
     |     |     +--rw sip-serv-id              uint8
     |     |     +--rw sip-serv-domain-name     string
     |     |     +--rw sip-serv-addr            inet:ipv6-address
     |     +--rw dns-config-option
     |     |  +--rw enable                      boolean
     |     |  +--rw dns-server* [dns-serv-id]
     |     |  |  +--rw dns-serv-id              uint8
     |     |  |  +--rw dns-serv-addr            inet:ipv6-address
     |     |  +--rw domain-search-list          string
     |     +--rw nis-config-option
     |     |  +--rw enable                      boolean
```

- **option-sets**: All currently standardized DHCPv6 options modeled (server)
- **option-set**: Specific set of options and their values to be provisioned to clients
- enable: a **boolean** node to indicate whether the option is valid in current set

Open issues

- How the model maps to specific implementations?
- How to be extensible for future work in IETF or new implementations?
- Are there areas where model can be improved?
- Hackathon in Yokohama? Any interests?

Next Steps

- Need to check how well it can fix existing implementations
- Plan to map into existing implementations:
  - Nominum DHCP
  - ISC DHCP
  - Kea
  - Huawei DHCP
  - … Any other implementations have interests?
- Model extensibility: plan to work when get familiar with existing implementations
- Is the WG interested in YANG work?
  - Yes -> adopt?
  - Maybe->continue working on this, reevaluate later
  - No ->drop the work

(backup)Motivation

- No unified method to configure DHCPv6 entities
  - Manual pre-configure should be discarded
- DHCPv6 employs multiple options for extensions
  - Not only for address allocation, support various kinds of configurations (e.g. DNS, SIP and etc)
  - Not easy to configure and manage
- NETCONF and YANG is flexible and extensible
  - Widely accepted in IETF
  - Appropriate for configuration and management of DHCPv6 components

(bk)DHCPv6 Server YANG Snippet

```
+--rw network-ranges
|  +--rw option-set [option-set-id]
|  +--rw network-range* […]
|     +--rw network-range-id
|     +--…
|     +--rw inherit-option-set
|     +--rw option-set [option-set-id]
|     +--rw address-pools
|     |  +--rw address-pool* [pool-id]
|     |  |  +--rw inherit-option-set
|     |  |  +--rw option-set […]
|     |  +--ro binding-info* [cli-id]
|     +--rw prefix-pools
|     +--rw hosts
+--rw relay-opaque-paras
|  +--rw relays* [relay-name]
|     +--rw relay-name
|     +--rw interface-info* [if-name]
|     +--…
+--rw rsoo-enabled-options
|  +--…
+--ro packet-stats
   +--ro solicit-count
```

- **hierarchy**: configuring at different levels **(global, network, pool & client)**
- **option inheritance**: option set in higher level is included in lower levels
- **inherit-option-set**: a **boolean** node to stop option inheritance
- opaque values in **Relay Agent options,** configured at server
- a configurable list of **rsoo-enabled-options**

(backup)DHCPv6 Relay Portion

```
+--rw relay
   +--rw relay-attributes
   |  +--…
   |  +--rw dest-addrs*
   |  +--rw subscribers* […]
   |  +--rw remote-host* […]
   |  +--rw vendor-info
   +--rw relay-supplied-options-option
   |  +--rw rsoo-set* [rsoo-set-id]
   |     +--rw erp-local-domain-name
   |           -option
   +--rw relay-interfaces
   |  +--rw relay-if* [if-name]
   |     +--rw if-name
   |     +--rw enable
   |     +--rw interface-id?
   |     +--rw rsoo-set [rsoo-set-id]
   |     +--rw pd-route* [pd-route-id]
   |     +--rw next-entity* [dest-addr]
   |        +--rw dest-addr
   |        +--rw available
   |        +--rw multicast
   |        +--rw server
   |        +--ro packet-stats
   +--ro relay-stats
      +--…
```

- Relay Agent options
- a set of options that need to be **configured at the relay**
- selected **rsoo option set** for this interface
- the **route** for delegated prefixes into edge router

(backup)DHCPv6 Client Portion

```
+--rw client
   +--rw client-interfaces
      +--rw client-if* [if-name]
         +--rw if-name
         +--…
         +--rw oro-options
         |  +--rw oro-option* […]
         |     +--…
         +--rw client-configured-options
         |  +--rw user-class-option
         |     +--rw enable
         |     +--…
         +--ro identity-associations
         |  +--ro identity-assoc
         |        iation*[iaid]
         |     +--ro iaid
         |     +--…
         +--ro if-other-paras
         |  +--…
         |  +--ro dns-paras
         |  +--…
         +--ro supported-options
         +--ro packet-stats
```

- **per-interface** manner
- options request in **ORO**
- **all possible options** need to be configured at client
- **IAs** obtained through DHCPv6
- **Extra** configuration data
- **State data** that declares which options are supported by client

(backup)DHCPv6 Notifications

```
+--n notifications
   +--n dhcpv6-server-event
   |  +--n pool-running-out
   |  |  +--n …
   |  +--n invalid-client-detected
   |     +--n …
   |
   +--n dhcpv6-relay-event
   |  +--n topo-changed
   |     +--n …
   |
   +--n dhcpv6-client-event
      +--n ia-lease-event
      |  +--n …
      +--n invalid-ia-detected
      |  +--n …
      +--n retransmission-failed
      |  +--n …
      +--n failed-status-turn-up
         +--n …
```

- addr/prefix pool is going to **run out**
- a client which can be regarded as an **attacker** is found
- **topology** of the relay agent is changed
- **allocated/rebind/renew/ release** an IA
- **invalid** IA is detected
- retransmission mechanism **failed**
- received message includes **failed status code**