I2RS protocol Design team

Attendees:

- Mahesh Jethanandani
- Andy Bierman
- Ignas Bogdonas,
- Jeff Haas,
- Anu Nair,
- Eric Voit

Discussion:

- Purpose of this group
    - The goal is to revise the strawman draft during the next week, and hand it over to NETCONF/RESTCONF experts to make sure the NETCONF/RESTCONF protocol is working correct.
    - Sue Hares has taken the editor's pen to work on this draft.
    - There will be lots of email in the next week

Topics discussed today:

- Ephemeral Database is only being defined in the I2RS work
    - Open Config requirements is using
        - intended configuration – this data represents the state that the network operator intends the system to be in. This data is colloquially referred to as the 'configuration' of the system.
        - applied configuration – this data represents the state that the network element is actually in, i.e., that which is currently being run by particular software modules (e.g., the BGP daemon), or other systems within the device (e.g., a secondary control- plane, or line card).
        - derived state - this data represents information which is generated as part of the system's own interactions. For example, derived state may consist of the results of protocol interactions (the negotiated duplex state of an Ethernet link), statistics (such as message queue depth), or counters (such as packet input or output bytes).
    - Open Configuration requirements are using the
- Routing Overlap
    - Less specific prefix versus more specific
        - CIDR suggests longer-prefix with shorter priority
        - Example 1 – both installed
            - 192.15/10/24 – priority 8

- 192.15/16 – priority 9
- Example 2 – same priority, same key, different next hops
  - 192.15.10/24 next-hop 192.1.1.1 priority 8
  - 192.15.10/24 nexthop 192.1.1.2 priority 8
  - First one takes precedence
- I2RS RIB has next-hop chain to take care of ECMP. The ECMP work should be buffered by the client that does the write.
- Current CLI does multiple writes for a next-hop and allows that to become ECMP, but the application can turn this into next-hop chain
- Note: Implementation may keep prefixes and next-hops, but change the link.

- Issue: How do we get active route indication?
  - Do we used the concept of applied configuration and derived configuration to clarify the ontext?

- Can there be multiple intents in the panes of glass model? (Eric)
  - Yes – the different intents can represent different applications
  - Example – I2RS DDOS client and I2RS configuration
    - I2RS DDOS 192.15.10/24 nexthop 192.1.1.2 priority 9
    - I2RS config 192.15.10/24 nexthop-chain (ecmp flag on) 192.1.1.1, 192.1.1.3, 192.1.1.4, 192.1.1.5
    - Nexthop 192.1.1.2 – goes to the DDOS filtering box
- Syntax checks
  - The check on a single variables range is constrained, but the checks on the cross-module contextual syntax is more difficult.
  - Example:
    - Global Flag RMON plus Other flags that look at RMON flag to determine if their tracing is on.
  - Result: Client will have to know the data tree to make things easier.
    - Yang module can provide a place to store specific flags
- Initial Implementations
  - RESTCONF – with patch can work through all I@RS
  - NETCONF – does not have an immediately workable situations
- Error handling on the following cases need to be worked out.
  - **The plan for bringing this to conclusion:** Sue will provide examples for routes, FB-RIB, Topology, BGP, ISIS, and OSPF

  Discussion:

  - 3 choices of collision:
    - No collision – must be accepted (accept and send notification)
    - Partial overlap with better priority (needs to be worked out)

        - Complete overlap with better data priority (accept and send notification)
    - Three choices of Action
        - All or nothing – (many implementations of RESTCONF or other provide the roll back internally)
        - Continue on Error – (Joel Halpern/Anu Nair will help with this one)
        - Stop on error –
    - Actual code working
        - RESTCONF – HTTP patch – must be atomic (all-or now)
            - Variable syntax can be done easily
            - Contextual syntax (module 1 + other modules) – is more difficult but can be done by the client if they work)
            - Resources syntax – do we have enough resources – this requires the client to send the response back
    - Mahesh: Operators want roll-back-on error (RESTCONF all-or-nothing) just as RESTCONF DOS
    - Jeff: the panes of glass within JUNOS ephemeral – sends directly to candidate and not to data store. The sending to datastore is holdover from IOS.
        - It may be there are embedded devices that use the rules?
        - Do we want to continue with the IOS or simply state the client must handle it
    - Andy:
        - Making it optional on the client and mandatory on the agent is backwards?
        - The agent needs to be simple with few options.
    - Jeff: I think we can life with an atomic commit (All or nothing).
    - Sue: We need to dig into the Continue on Error – with Joel. I will send notes to Joel and Anu to work through this point.
    - Andy: What happens when you do an edit? Does the server accept the edit and produce the appropriate information
    - Jeff: Dynamic datastore in Juniper. The overlap between multiple panes works through these issues.
        - Policy issue: if you configure BGP policy, but the BGP peer is down that use the policy – then the policy does work.
    - Andy – you are writing to the special data store of “ephemeral” intended config?

- Priority – is there maximum priority the same as the maximum clients or is it a range?
    - Sue: Currently it is a range. Kent proposed a maximum priority
    - Example of range:
        - 1-32 range with 100 clients (too many clients)
        - 1-100 with 10 clients (too few clients)
    - The plan had been to store client ID and then to link priority off that client ID.

    - If the client ID changes, then the I2RS agent must change the process.
- Alignment of Routing RIBG and I2RS RIB
    - Alignment is not possible due to the different semantic for next-hops.
    - We can try to align it as much as possible.
- Batch update
    - What is rate:
        - 5000/second or 250,000 per second (validation done outside the processes in core).
        - CLI currently does 10/second
    - What happens if 80% of batch succeeds and 20% fail
        - I2RS needs a programmatic input on fail
        - Flags to indicate processing
            - To be checked: Should we allow "no validation" as option with the caveat that I2RS agent must beware
            - To be checked: Is an RPC better than this put
    - Should it be RPC or PUT to data store?
        - Anu: Both have a p lace. RPC will handle Batch to application and it decides what to do with the overhead. In a large batch file this will work.
        - PUT – sends it directly to the data store.
    - RESTCONF Edit – can set a full set of subtrees in a single patch (Andy)
        - Does batching mean changing sub-trees?
        - Jeff: For 250,000 updates to Routes, you are working in the same subtree.
        - Andy: One wild example is 100K interfaces in a single back (probably sent by a QA department).
    - Resolved:
        - Sue will propose RPC and PUT batch module to the list
- Config False nodes
    - Panes of glass options (Jeff)
        - Leave the pane of glass there, and let I2RS agent resolve installation
        - Create a mechanism to ask for active configured node
        - We need active-config (from openconf opstate) as well as derived state (route nexthops that are active due to interface up)
        - This makes it complicated.
    - Active config is useful for epehemral database.
- What about traceability? (Andy)
    - (jeff) – Traceability is the wrong hammer to put in the response for nodes. We have not reached the point in the protocol to determine if traceability has the right information.
    - (Sue): I will review traceability this week in preparing the document.