NETCONF Server and RESTCONF Server Configuration Models

draft-ietf-netconf-server-model-06

NETCONF WG
IETF #92 Dallas, TX, USA

Updates since IETF 90

- Removed YANG 1.1 style if-feature statements
- Removed the read-only lists of SSH host-keys and TLS certs
- Added ability to configure trust-anchors for SSH X.509 client certs
- Now imports by revision, per best practice (?)
- Added support for RESTCONF server
- Added RFC Editor instructions
- Added NACM statements to YANG modules for sensitive nodes
- Added client-cert-auth subtree to ietf-restconf-server module
- Added description for braces to tree diagram section
- Renamed feature from "rfc6187" to "ssh-x509-certs"

Last Call Results

- Model changes needed
- Some editorial clarifications needed

Open Issues

https://github.com/netconf-wg/server-model/issues

#32: rename "application" node name to "netconf-client”?

- Current:

```
module: ietf-netconf-server
   +--rw netconf-server
      +--rw call-home {call-home}?
         +--rw application* [name]
            +--rw ssh
               +--rw endpoints
                  +--rw endpoint* [name]
                     ...
```

- Proposed:

```
module: ietf-netconf-server
   +--rw netconf-server
      +--rw call-home {call-home}?
         +--rw netconf-client* [name]
            +--rw ssh
               +--rw endpoints
                  +--rw endpoint* [name]
                     ...
```

#33: Is it a good idea to name the top-level node "netconf-server"?

- Is this name consistent with how we name other things?
  - what might be better?
  - FWIW, RFC 6022 has "netconf-state”
- Example using current naming strategy:

```
<netconf-server
    xmlns="urn:ietf:params:xml:ns:yang:ietf-netconf-server">
    <session-options>...</session-options>
    <listen>...</listen>

    <call-home>...</call-home>
    <ssh>...</ssh>
</netconf-server>
```

#34: Are the current features granular enough?

- For NETCONF only, it's not possible to advertise being able to listen for just SSH and call-home with just TLS

```
+--rw netconf-server
        +--rw listen {listen}?
        |  +--rw endpoint* [name]
        |     +--rw (transport)
        |        +--:(ssh) {ssh}?
        |        +--:(tls) {tls}?
        +--rw call-home {call-home}?
        |        +--rw application* [name]
        |        +--rw (transport)
        |           +--:(ssh) {ssh}?
        |           +--:(tls) {tls}?
        +--rw ssh {ssh}?
           ...
        +--rw tls {tls}?
           ...
```

YANG 1.1's new if-feature syntax was designed to support this case, but can't use because 6020bis isn't stable yet…

#36: is import by revision needed?

- At the time I submitted this draft, it was my understanding that import by revision was best practice, and that prior YANG modules were in violation.
- Recent YANG 1.1 conformance discussions seem to be swinging the other direction now, but with potential to swing back again.
- Unclear what the *right* thing to do is !
- Perhaps taking it out is the way to go because, even if it's wrong, it will at least be in the company of other published modules ;)

#38: remove upper-bound on hello-timeout, idle-timeout, and max-sessions?

```
leaf hello-timeout {
   type uint32 {
      range "0 | 10 .. 3600";
   }
   units "seconds";
}

leaf idle-timeout {
   type uint32 {
      range "0 | 10 .. 360000";
   }
   units "seconds";
}

leaf max-sessions {
   type uint16 {
      range "0 .. 1024";
   }
}
```

#39: move away from a number with a fixed unit?

- Removing upper-bounds is well and good, but large values can become unreadable:
  - Example: 3 days or 259,200 seconds?
- How about a 2-tuple?
  - One leaf for a numerical value
  - One leaf for an enum [secs, mins, hours, days, etc.]
- Or something like a XSD's "duration"?
  - Example: PnYnMnDTnHnMnS

#40: move "max-sessions" to global session-param?

- Currently under the “listen” leaf
- If moved to global level, how to catch if configured number of call-homes exceed the value?
- Can an must statement catch this?
  - count(/call-home/application) <= /session-options/max-sessions

#41: should address be mandatory?

- Currently, neither address nor port are mandatory for a listening endpoint
  - but port has a default

- should address be mandatory
  - or should no specified address be treated as a wildcard?

#43: keep-alive, linger, reconnect interval defaults OK?

- …/connection-type/persistent/keep-alives/interval-secs:
  - 15 seconds
- …/connection-type/periodic/linger-secs:
  - 30 seconds
- …/reconnect-strategy/interval-secs:
  - 5 minutes

#45: how do interval-secs and count-max work for reconnect-strategy if an endpoint resolves to multiple IP addresses?

- E.g., let's say an application has 3 endpoints
  - name1, name2, and name3
- And each expands into two IP addresses:
  - {ip1.1, ip1.2}, {ip2.1, ip2.2}, {ip3.1, ip3.2}
- Proposal: treat as if IPs were configured explicitly
  - E.g., ip1.1 → ip1.2 → ip2.1 → ip2.2 → ip3.1 → ip3.2

#46: move "peer_allowed_to_send" to CH draft?

- Currently Call Home draft says nothing about keep alives!
  - It should say “Servers SHOULD send keep-alives…”
- But in order to do so, TLS [RFC 6520] requires the client to advertise "peer_allowed_to_send”
  - Thus we also need “Clients MUST advertise "peer_allowed_to_send"
- Proposal: move entire Section 5 to Call Home draft
  - Section 5: Implementation strategy for keep-alives
    - Covers both SSH and TLS keep-alives

#47: introduce a 2nd timeout for periodic connections for when there's data to send?

- Current text says that a device SHOULD pro-actively connect to the client if it has messages to send

- Options:
  1. Leave as it is
  2. have another configurable timer (less than periodic interval) for how long device should wait?
  3. Or an absolute time (e.g., 2:00am) ?

#49: combine trusted-ca-certs and trusted-client-certs for ssh/tls?

-
  and client-certs for SSH and TLS
- There doesn’t seem to be a Security reason for why these are separate
- Would like to combine, but how to set if-feature statement?
- Ideally would use YANG 1.1 if-feature syntax
  - if-feature “(ssh-x509-certs or tls)”;
- Create feature called “ssh-x509-or-tls”?

Next Steps

- Another Last Call will be necessary

Thank you