

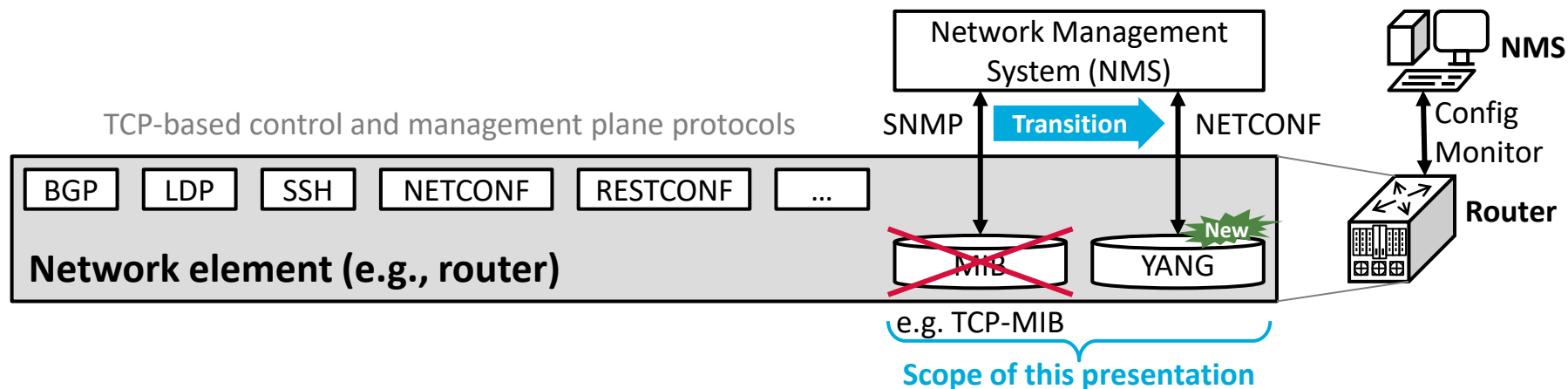
Transmission Control Protocol (TCP) YANG Model draft-scharf-tcpm-yang-tcp

Michael Scharf <michael.scharf@hs-esslingen.de>

March 25, 2019

Michael Scharf is supported by the StandICT.eu project, which is funded by the European Commission under the Horizon 2020 Programme

TCP on embedded network devices



- TCP used by many **control and management plane protocols** (such as BGP)
 - Requires (some) TCP stack configuration and monitoring on network elements (e.g., routers)
 - Network management traditionally by CLI or SNMP
 - Implementations of TCP-MIB (RFC 4022) apparently exist
- **Ongoing transition from SNMP/MIB to NETCONF/YANG**
 - MIB-based YANG models generated by the algorithm in RFC 6643 as remedy?
 - Example: <https://github.com/YangModels/yang/tree/master/vendor/cisco/xe>
- **Disclaimer:** Potential differences to TCP stacks in operating systems of Internet hosts

Quotation from draft-scharf-tcpm-yang-tcp: “Many protocol stacks on Internet hosts use other methods to configure TCP, such as operating system configuration or policies. Such TCP stacks typically cannot be configured by network management protocols such as NETCONF or RESTCONF. **This document only applies to network elements that use YANG data models.**”

YANG model auto-generated from TCP-MIB

```
module: TCP-MIB
  +--rw tcp
    +--ro tcpRtoAlgorithm?      enumeration
    +--ro tcpRtoMin?           int32
    +--ro tcpRtoMax?           int32
    +--ro tcpMaxConn?          int32
    +--ro tcpActiveOpens?      yang:counter32
    +--ro tcpPassiveOpens?     yang:counter32
    +--ro tcpAttemptFails?     yang:counter32
    +--ro tcpEstabResets?      yang:counter32
    +--ro tcpCurrEstab?        yang:gauge32
    +--ro tcpInSegs?           yang:counter32
    +--ro tcpOutSegs?          yang:counter32
    +--ro tcpRetransSegs?      yang:counter32
  x--rw tcpConnEntry* [tcpConnLocalAddress tcpConnLocalPort tcpConnRemAddress tcpConnRemPort]
    | x--rw tcpConnState?      enumeration
    | x--ro tcpConnLocalAddress inet:ipv4-address
    | x--ro tcpConnLocalPort   int32
    | x--ro tcpConnRemAddress  inet:ipv4-address
    | x--ro tcpConnRemPort     int32
    +--ro tcpInErrs?           yang:counter32
    +--ro tcpOutRsts?          yang:counter32
    +--ro tcpHCInSegs?         yang:counter64
    +--ro tcpHCOutSegs?        yang:counter64
  +--rw tcpConnectionEntry* [tcpConnectionLocalAddressType tcpConnectionLocalAddress \
tcpConnectionLocalPort tcpConnectionRemAddressType tcpConnectionRemAddress tcpConnectionRemPort]
    | +--ro tcpConnectionLocalAddressType inet-address:InetAddressType
    | +--ro tcpConnectionLocalAddress   inet-address:InetAddress
    | +--ro tcpConnectionLocalPort      inet-address:InetPortNumber
    | +--ro tcpConnectionRemAddressType  inet-address:InetAddressType
    | +--ro tcpConnectionRemAddress     inet-address:InetAddress
    | +--ro tcpConnectionRemPort        inet-address:InetPortNumber
    | +--rw tcpConnectionState?         enumeration
    | +--ro tcpConnectionProcess?       uint32
  +--rw tcpListenerEntry* [tcpListenerLocalAddressType tcpListenerLocalAddress tcpListenerLocalPort]
    +--ro tcpListenerLocalAddressType  inet-address:InetAddressType
    +--ro tcpListenerLocalAddress       inet-address:InetAddress
    +--ro tcpListenerLocalPort         inet-address:InetPortNumber
    +--ro tcpListenerProcess?          uint32
```

Generated by smidump 0.4.8 library

TCP configuration
(only for RTO, outdated!)

TCP stats
(mostly counters)

TCP connection table
(actually writable
according to RFC 4022!)

TCP listener table

Transmission Control Protocol (TCP) YANG Model

- **draft-scharf-tcpm-yang-tcp-01** mostly as a placeholder
 - Further work and possibly more authors needed ;-)
 - YANG model auto-generated from TCP-MIB only to enable change tracking
 - TCPM charter explicitly includes “interfaces that maintain TCP's utility”
 - Note: IETF YANG models are typically standards track
- **Questions if (and only if) follow-up versions shall be published**
 - Scope: What TCP parameters shall be included?
 - Backward compatibility: How closely shall it align with the TCP-MIB?
 - Extended statistics: Shall it deal with extended statistics (RFC 4898)?
 - Suggested MIB additions: Consider TCP-MIB extensions such as RFC 5482?
 - ... and more (see document)
- **Potential next steps**
 1. Do nothing
 2. Minimal TCP YANG model very similar to TCP-MIB
 3. More comprehensive YANG model with “common” TCP config (e.g. on routers)
 4. Try to boil the ocean...