# Using YANG and NETCONF for LMAP

Jürgen Schönwälder

IETF 86, Orlando, 2013-03-13

# YANG and NETCONF

## NETCONF

The Network Configuration Protocol (NETCONF) provides mechanisms to install, manipulate, and delete the configuration of network devices. It uses an XML-based encoding of the configuration data as well as the protocol messages. The NETCONF protocol operations are realized as remote procedure calls (RPCs). The NETCONF protocol runs by default over SSH but it can also be used over TLS with pairwise authentication using X.509 certificates.

## YANG

YANG is a data modeling language used to model configuration and state data manipulated by NETCONF, NETCONF remote procedure calls, and NETCONF notifications.

# Sketch of a YANG Data Model: Tests

The data model has the following structure, where square brackets are used to enclose a list's keys, "?" means that the leaf is optional, and "*" denotes a leaf-list:

```
module: acme-lmap
   +--rw lmap
      +--rw tests
      |  +--rw test [name]
      |     +--rw name           string
      |     +--rw description?   string
      |     +--rw program        string
      |     +--rw option [name]
      |     |  +--rw name        string
      |     |  +--rw value?      string
      |     +--rw argument*      string
```

# Sketch of a YANG Data Model: Scheduling

```
+--rw schedules
      +--rw test?           leafref
      +--rw enabled?        boolean
      +--rw (schedule-type)?
      |  +--:(periodic)
      |  |  +--rw interval?       uint32
      |  +--:(calendar)
      |  |  +--rw weekday?        weekday-set
      |  |  +--rw month-set?      months-set
      |  |  +--rw day*            int8
      |  |  +--rw hour*           int8
      |  |  +--rw minute*         int8
      |  +--:(one-shot)
      +--ro failures?       yang:counter32
      +--ro last-failure?   string
      +--ro last-failed?    yang:date-and-time
```

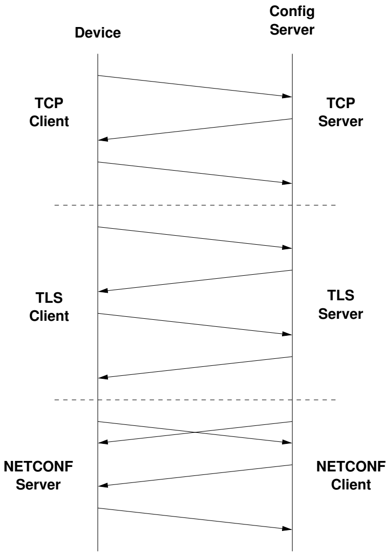
# Properties of Measurement Agents

1. MAs are often deployed behind Network Address Translators (NATs). This might even be true if MAs are part of a device on the demarcation line between a service provider and a home network due to the usage of Carried Grade NATs in the service provider network.
2. MAs may run on devices that are not always powered up and online.
3. A single controller may be responsible for a large number of MAs.
4. A large fraction of the MAs may be inactive (i.e., they do not perform any measurements) at any given point in time. Inactive MAs may need to be enabled on demand to troubleshoot specific problems (e.g., as part of customer helpdesk services) or to balance measurement traffic load.

# NETCONF Issue #1: Connection Initiation

- NETCONF has been originally designed to be used on network devices such as backbone routers. A device supporting NETCONF has an embedded NETCONF server.
- Configuration management applications use embedded NETCONF clients to connect to NETCONF servers and then issue RPC calls to manipulate the configuration state of the devices.
- Due to the nature of LMAP MAs (likely located behind NATs), it is crucial that MAs initiate connections to a controller. This is currently not supported in NETCONF.

# NETCONF Issue #1: Call Home for TLS



1. Device initiates TCP connection (based on a certain schedule)
2. Device initiates TLS exchange with pairwise X.509 authentication
3. Device hands connection over to a NETCONF server, config server hands over to a NETCONF client
4. NETCONF `<hello>` exchange ensures proper roles are picked

# NETCONF Issue #2: Configuration Versions

## Issue

The controller (running a NETCONF client) must determine whether a device's configuration needs updates. While this could be achieved by retrieving the configuration using `<get-config>` and comparing the result with the expected configuration, this approach is not very efficient.

## Proposal

The NETCONF indicates the version of the configuration it is currently using. (The version can either be identified by a version number or a time-stamp of the last configuration change or simply an opaque tag that is handed out and interpreted only by the controller.) As an optimization, the configuration version might be carried as a capability in the `<hello>` exchange.

# NETCONF Issue #3: Pushing of Measurement Results

## Issue

NETCONF has not been designed as a data push protocol. While a NETCONF extension provides support for event notifications, this mechanism requires in its simplest form that a NETCONF client first subscribes to an event stream and that the session used to carry event notifications stays open.

- Option #1: Make use of the event notification replay feature: A MA is locally collecting measurement results. After connecting to a collector (acting as a NETCONF client), the collector subscribes to an event stream with a request to replay the measurement results collected since the last time data has been fetched from the MA.
- Option #2: Model test results as part of an LMAP data model and use NETCONF <get> operations to retrieve the data and a new RPC to clear the data.

# Discussion

## Implementation and Deployment Considerations

- Typical platforms used for building hardware probes and home routers usually do not support NETCONF today
- Memory impact of NETCONF servers not fully understood (we compiled `libnetconf` to run on OpenWrt platforms and we had to remove a number of unnecessary libraries)
- Configuration server logic would need to be adapted to clients calling home

## Alternatives

- Follow the pattern that current large scale measurement systems are using: REST APIs over HTTPs with the MA acting as an HTTP client
- A simple standard `curl` program on the measurement agent is sufficient to get the protocol work done

# References

M. Bjorklund.
YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF).
RFC 6020, Tail-f Systems, October 2010.

J. Schönwälder.
Common YANG Data Types.
RFC 6021, Jacobs University, October 2010.

R. Enns, M. Bjorklund, J. Schönwälder, and A. Bierman.
Network Configuration Protocol (NETCONF).
RFC 6241, Juniper Networks, Tail-f Systems, Jacobs University, Brocade, June 2011.

M. Wasserman.
Using the NETCONF Protocol over Secure Shell (SSH).
RFC 6242, Painless Security, June 2011.

M. Badra.
NETCONF over Transport Layer Security (TLS).
RFC 5539, CNRS/LIMOS Laboratory, May 2009.

J. Schönwälder.
A YANG Data Model for LMAP Measurement Agents.
Internet Draft (work in progress) <draft-schoenw-lmap-yang-00.txt>, Jacobs University, February 2013.

J. Schönwälder.
Considerations on using NETCONF with LMAP Measurement Agents.
Internet Draft (work in progress) <draft-schoenw-lmap-netconf-00.txt>, Jacobs University, February 2013.

D. Levi and J. Schönwälder.
Definitions of Managed Objects for Scheduling Management Operations.
RFC 2591, Nortel Networks, TU Braunschweig, May 1999.