                 Using RESTCONF with LMAP Measurement Agents
                    draft-schoenw-lmap-restconf-00.txt

Abstract

   This document describes how RESTCONF can be used with a YANG data
   model for Large-Scale Measurement Platforms (LMAP).

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

   This document discusses how a controller can use the RESTCONF
   protocol [I-D.ietf-netconf-restconf] to configure Large-Scale
   Measurement of Broadband Performance (LMAP) measurement agents (MAs)
   [I-D.ietf-lmap-framework]. It also discusses how RESTCONF can be
   used to report measurement results to a collector.

   MAs may be deployed as separate hardware devices or as functions
   embedded in consumer electronic devices and home routers or as pure
   software solutions that can be installed on off-the-shelf computing
   equipment. Measurement agents receive instructions from a controller
   about when and how to conduct what measurements (the measurement
   schedule) and how and when to report measurement results to a data
   collector (the report schedule). Further information about the
   interaction between MAs and controllers and collectors can be found
   in [I-D.ietf-lmap-framework].

   The LMAP information model [I-D.ietf-lmap-information-model] defines
   the information exchanged between a controller and an MA and the
   information exchanged between an MA and a collector. An information
   model is conceptual and protocol-independent. A concrete YANG
   [RFC6020] data model derived from the conceptual information model is
   defined in [I-D.schoenw-lmap-yang].

2. Overview of RESTCONF

   The RESTCONF protocol [I-D.ietf-netconf-restconf] provides a REST-
   like interface to access and manipulate a so-called unified YANG
   datastore [RFC6020]. The basic idea behind RESTCONF is expose a YANG
   datastores as a collection of Web resources that can be manipulated
   using standard HTTP [RFC7230] DELETE, PATCH, POST, and PUT methods.

The resource hierarchy is derived from the nesting structure of the
YANG schema tree, leading to a so called data model driven REST API.

RESTCONF is essentially a convention how to use HTTP over TLS to
access a datastore that has a structure defined by a YANG data model.
The data is exchanged in XML encoding or JSON encoding.

The normal mode of operation is that the RESTCONF client initiates a
secure transport to the RESTCONF server. For devices located behind
a NAT, a so called 'call-home' mechanism has been defined
[I-D.ietf-netconf-call-home] that enables the RESTCONF server to
establish a secure transport to a RESTCONF client. Note that call
home only changes the TCP connection establishment, the TLS and HTTP
client/server roles do not change. The policy used to call home can
be configured through a configuration data model
[I-D.ietf-netconf-server-model]. This model provides mechanism to
configure a list of redundant endpoints and it provides control over
call-home policies (e.g, call-home frequency, idle-timers, keep-alive
timers).

3. RESTCONF as LMAP Control Protocol

It is straight-forward to user RESTCONF as a control protocol. The
YANG data model [I-D.schoenw-lmap-yang] derived from the underlying
information model [I-D.ietf-lmap-information-model] translates into a
collection of RESTCONF resources that can be manipulated at various
levels of granularity using DELETE, PATCH, POST, and PUT methods.

An example exchange showing a REST call to create a schedule object
is shown in Appendix B.

4. RESTCONF as LMAP Report Protocol

One way of mapping the information model parts relevant for reports
into a YANG data model is the usage of YANG notifications. This is
the approach currently used in [I-D.schoenw-lmap-yang]. This mapping
leads to report notifications that push measurement results from the
MA to a collector. The RESTCONF protocol uses Server Sent Events as
the underlying mechanism to stream notifications.

A direct mapping of the information model leads to relatively verbose
exchanges and it is possible to define more space efficient
notifications that suppress information that is only changing rarely.

An example exchange of a result notification is shown in Appendix C.

Note that alternative designs are possible. One option is to make
the collector a RESTCONF server and to have the MA push results to

the collector by posting to resources on the controller. Another
option is to have the results reside on the MA and to export the
results as part of the operational state of the MA. The collector(s)
will then use GET requests to fetch the result resources from the MA.
Note that all three approaches can be implemented using RESTCONF and
YANG.

5. Conclusion

This document discusses how RESTCONF can be used as a control
protocol and a report protocol for Large-Scale Measurement Platforms
(LMAP). The benefit of using RESTCONF is that no new protocol work
needs to be done. Additional benefits derive from using YANG and a
data model driven approach. Despite the reuse of existing tools,
using a data model driven approach allows to easily reuse other data
models (e.g., network interfaces [RFC7223], [RFC7277] or general
system services [RFC7317]) in order to export additional status
information about an MA to a controller.

6. IANA Considerations

This document has no requests for IANA.

7. Acknowledgements

Juergen Schoenwaelder and Vaibhav Bajpai work in part on the Leone
research project, which receives funding from the European Union
Seventh Framework Programme [FP7/2007-2013] under grant agreement
number 317647.

8. Informative References

[I-D.ietf-lmap-framework]
          Eardley, P., Morton, A., Bagnulo, M., Burbridge, T.,
          Aitken, P., and A. Akhter, "A framework for Large-Scale
          Measurement of Broadband Performance (LMAP)", draft-ietf-
          lmap-framework-12 (work in progress), March 2015.

[I-D.ietf-lmap-information-model]
          Burbridge, T., Eardley, P., Bagnulo, M., and J.
          Schoenwaelder, "Information Model for Large-Scale
          Measurement Platforms (LMAP)", draft-ietf-lmap-
          information-model-04 (work in progress), March 2015.

[I-D.ietf-netconf-call-home]
          Watsen, K., "NETCONF Call Home and RESTCONF Call Home",
          draft-ietf-netconf-call-home-04 (work in progress),
          February 2015.

    [I-D.ietf-netconf-restconf]
              Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF
              Protocol", draft-ietf-netconf-restconf-04 (work in
              progress), January 2015.

    [I-D.ietf-netconf-server-model]
              Watsen, K. and J. Schoenwaelder, "NETCONF Server and
              RESTCONF Server Configuration Models", draft-ietf-netconf-
              server-model-06 (work in progress), February 2015.

    [I-D.schoenw-lmap-yang]
              Schoenwaelder, J. and V. Bajpai, "A YANG Data Model for
              LMAP Measurement Agents", draft-schoenw-lmap-yang-02 (work
              in progress), January 2015.

    [I-D.starkcarey-lmap-protocol-criteria]
              Stark, B. and T. Carey, "LMAP Protocol Selection
              Criteria", draft-starkcarey-lmap-protocol-criteria-01
              (work in progress), March 2015.

    [RFC6020] Bjorklund, M., "YANG - A Data Modeling Language for the
              Network Configuration Protocol (NETCONF)", RFC 6020,
              October 2010.

    [RFC7223] Bjorklund, M., "A YANG Data Model for Interface
              Management", RFC 7223, May 2014.

    [RFC7230] Fielding, R. and J. Reschke, "Hypertext Transfer Protocol
              (HTTP/1.1): Message Syntax and Routing", RFC 7230, June
              2014.

    [RFC7277] Bjorklund, M., "A YANG Data Model for IP Management", RFC
              7277, June 2014.

    [RFC7317] Bierman, A. and M. Bjorklund, "A YANG Data Model for
              System Management", RFC 7317, August 2014.

Appendix A. Response to Protocol Comparison Criteria

   A set of control and report protocol comparison criteria has been
   defined in [I-D.starkcarey-lmap-protocol-criteria]. This section
   compares the usage of RESTCONF against the criteria.

   CP-MUST-1   yes (RFC6241 RFC6242, RFC5539),
               [I-D.ietf-netconf-restconf]

   CP-MUST-2   yes [I-D.ietf-netconf-call-home]

CP-MUST-3   yes SSH / TLS (NETCONF), TLS (RESTCONF)

CP-MUST-4   yes YANG data models [RFC6020] have a well defined
            versioning and extension model

CP-DIFF-1   1

CP-DIFF-2   1

CP-DIFF-3   yes

CP-DIFF-4   yes (NETCONF|RESTCONF) call home

CP-DIFF-5   (underspecified - it is JSON or XML over HTTP/TLS)

CP-DIFF-6   (underspecified - it is JSON or XML over HTTP/TLS)

CP-DIFF-7   HTTP and JSON/XML are pretty much everywhere

CP-DIFF-8   YANG tools are out there and the rest will develop, HTTP
            and TLS are pretty well understood

CP-DIFF-9   yes

CP-DIFF-10 many tools out there to create REST code, YANG tools
            available

CP-DIFF-11 yes, YANG RFC 6020 data models have a version model

CP-DIFF-12 additional YANG modules can augment the standard data
            model

CP-DIFF-13 JSON and XML, CBOR in the making

RP-MUST-1   yes [I-D.ietf-netconf-call-home]

RP-MUST-2   SSH / TLS (NETCONF), TLS (RESTCONF)

RP-MUST-3   YANG RFC 6020 data models have a version model

RP-DIFF-1   TCP

RP-DIFF-2   yes

RP-DIFF-3   (underspecified - it is HTTP over TLS)

RP-DIFF-4   yes

   RP-DIFF-5   yes

   RP-DIFF-6   yes (as part of HTTP encoding negotiations)

   RP-DIFF-7   (underspecified - it is JSON or XML over HTTP/TLS)

   RP-DIFF-8   HTTP and JSON/XML are pretty much everywhere

   RP-DIFF-9   many tools out there to create REST code, YANG tools
               available

   RP-DIFF-10 yes

   RP-DIFF-11 many tools out there to create REST code, YANG tools
               available

   RP-DIFF-12 JSON and XML, CBOR in the making

Appendix B. Example RESTCONF Control Protocol Exchange

   Below is a YANG tree diagram of a part of the data model covering
   schedules. This is taken from [I-D.schoenw-lmap-yang].

```
   module: ietf-lmap
      +--rw lmap
         +--rw schedules
            +--rw schedule* [name]
               +--rw name      string
               +--rw action* [name]
               |  +--rw name           string
               |  +--rw task           leafref
               |  +--rw option* [name]
               |  |  +--rw name     string
               |  |  +--rw value?   string
               |  +--rw destination* [name]
               |     +--rw name          string
               |     +--rw output*     uint16
               |     +--rw schedule    leafref
               |     +--rw action      leafref
               +--rw timing    leafref
```

   Below is an XML representation of instance data conforming to the
   YANG data model is shown below. Note that some of the strings are

references to other portions of the instance data not show here.
This is again taken from [I-D.schoenw-lmap-yang].

```
<lmap xmlns="urn:ietf:params:xml:ns:yang:ietf-lmap">
  <schedules>
    <schedule>
      <name>hourly</name>
      <action>
        <name>icmp-latency-hourly</name>
        <task>icmp-latency-measurement</task>
        <destination>
          <name>q-all</name>
          <schedule>daily</schedule>
          <action>report-daily</action>
        </destination>
      </action>
      <timing>hourly</timing>
    </schedule>
  </schedules>
</lmap>
```

Below is an example showing how RESTCONF can be used to create the
above schedule. The prefix C: indicates the controller, the prefix
M: indicates the measurement agent. This example uses a JSON
encoding (and note that much of the white-space can be removed, this
is only there to help with readability).

```
C: POST /restconf/data/ietf-lmap:lmap/schedules HTTP/1.1
C: Host: example.com
C: Content-Type: application/yang.data+json
C:
C:    {
C:      "ietf-lmap:schedule": {
C:        "name": "hourly",
C:        "action": [
C:           {
C:             "name": "icmp-latency-hourly",
C:             "task": "icmp-latency-measurement",
C:             "destination": [
C:               {
C:                 "name": "q-all",
C:                 "schedule": "daily",
C:                 "action": "report-daily"
C:               }
C:             ]
C:           }
C:        ],
C:        "timing": "hourly"
C:      }
C:    }

M: HTTP/1.1 201 Created
M: Date: Mon, 23 Apr 2012 17:01:00 GMT
M: Server: example-server
M: Location: https://example.com/restconf/data/ietf-lmap:lmap/schedules/
schedule=hourly
M: Last-Modified: Mon, 23 Apr 2012 17:01:00 GMT
M: ETag: b3a3e673be2
```

Appendix C. Example RESTCONF Report Protocol Exchange

The first step taken by the collector is to lookup the event stream
resource.

```
  C: GET /restconf/data/ietf-restconf-monitoring:restconf-state/streams/
stream=NETCONF/encoding=xml/events HTTP/1.1
  C: Host: example.com
  C: Accept: application/yang.data+xml

  M: HTTP/1.1 200 OK
  M: Content-Type: application/yang.api+xml
  M:
  M: <events
  M:  xmlns="urn:ietf:params:xml:ns:yang:ietf-restconf-monitoring">
  M:  https://example.com/streams/NETCONF
  M: </events>
```

Once the event stream resource is known (information might be
cached), the collector subscribes to the event stream resource.

```
      C: GET /streams/NETCONF HTTP/1.1
      C: Host: example.com
      C: Accept: text/event-stream
      C: Cache-Control: no-cache
      C: Connection: keep-alive

      M: data: <notification
      M: data:     xmlns="urn:ietf:params:xml:ns:yang:ietf-restconf">
      M: data:     <event-time>2013-12-21T00:01:00Z</event-time>
      M: data:     <report xmlns="urn:ietf:params:xml:ns:yang:ietf-lmap">
      M: data:     </report>
      M: data:

      M: data: {
      M: data:     "ietf-restconf:notification": {
      M: data:       "event-time": "2015-02-25T00:01:00Z",
      M: data:        "ietf-lmap:report": {
      M: data:           "date": "2015-02-25T00:01:00Z",
      M: data:           "agent-id": "xxx",
      M: data:           "header": {
      M: data:             "column": "target",
      M: data:             "column": "rtt"
      M: data:           }
      M: data:           "row": {
      M: data:             "start": "2015-02-25T00:01:00Z",
      M: data:             "value": "192.0.2.1",
      M: data:             "value": 42
      M: data:           }
      M: data:           "row": {
      M: data:             "start": "2015-02-25T00:01:00Z",
      M: data:             "value": "192.0.2.2",
      M: data:             "value": 24
      M: data:           }
      M: data:        }
      M: data:     }
      M: data: }
```

Authors' Addresses

Juergen Schoenwaelder
Jacobs University Bremen

Email: j.schoenwaelder@jacobs-university.de

   Vaibhav Bajpai
   Jacobs University Bremen

   Email: v.bajpai@jacobs-university.de