Interoperability Test of NETCONF Capabilities

D. Valencic and V. Pupovac
University of Applied Sciences Velika Gorica, Velika Gorica, Croatia
davorin_valencic@yahoo.com, vedran.pupovac@gmail.com

Abstract - **NETCONF is an important component of SDN (Software Defined Networking) architecture (as defined in RFC 7426) and network programmability and automation. ETF has defined the NETCONF (Network Configuration Protocol) standard in RFC 4741 in December 2006 and improved it in June 2011 in RFC 6241 (which is the currently active standard). NETCONF was created based on the requests of the network operators (defined in RFC 3535 in June 2002) for a standard protocol for the configuration of network devices. NETCONF defines the mechanisms of installing, manipulating, and deleting the network device configurations. NETCONF uses RPC messages, SSH for the transfer of messages between the server and the client; it has its defined set of operations and it uses YANG for data modelling. NETCONF defines the initial basic set of operations and additional capabilities. Capabilities are additional functions of the NETCONF device (of client and server) that may but do not have to be implemented by the network devices. In this paper will be shown an overview of the capabilities of NETCONF protocol. The objective of this paper is to present the results of the interoperability lab where the support for NETCONF capabilities have been tested on router/switch (R/S) devices of major network equipment manufacturers. This paper describes the test environment and how the laboratory testing of the NETCONF capabilities of virtual network R/S devices was performed.**

Keywords – NETCONF; interoperability; SDN; network programmability; automation

I. INTRODUCTION

In June 2002 Internet Architecture Board and the most important members of Internet Engineering Task Force (IETF) community for networking control held a meeting with the network operators. The results of the meeting have been documented in RFC 3535. It showed that the network operators were using CLI (Command line interface) user interfaces of network equipment manufacturers for the configuration of devices, i.e. console operating method and telnet/SSH protocols. The advantage of such operating method is that it is text-oriented, unlike BER (Basic Encoding Rules), i.e. ASN.1 coded method of Simple Network Management protocol (SNMP) [1].

In order to satisfy the requirements in RFC 3535 the first version of NETCONF protocol was developed, defined in RFC 4741 published in December 2006. NETCONF protocol was then improved in June 2011 in RFC 6241 [2] [3].

NETCONF provides the mechanism of installation, manipulation, and deletion of network devices configuration. NETCONF protocol is divided into four layers [6]:

- *secure transport layer*: provides secure and reliable transmission of messages between the client and the server.
- *messages layer*: describes the coding mechanism of remote procedure call (RPC) and notification.
- *operations layer*: defines a group of basic operations of the protocol that allow retrieval and editing of configuration data. Table I presents the basic operations of NETCONF protocol.
- *content layer*: consists of configuration data and notification data. NETCONF protocol does not deal with the contents layer in its RFC documents, but rather a separated protocol has been made, titled YANG. YANG is defined in RFC 6020, and RFC 6021 presents the types of data [2].

Basic NETCONF functionalities can be expanded by defining additional possibilities (NETCONF capabilities) while establishing the session between the client and the server. NETCONF allows a client to discover the set of protocol extensions supported by a server. These capabilities permit the client to adjust its behaviour to take advantage of the features exposed by the device. The capability definitions can be easily extended in noncentralized manner [3].

Capabilities are additional functions of the NETCONF

TABLE I. NETCONF OPERATIONS [3]

Operation	***Description***
<get>	Retrieve running configuration and device state information
<get-config>	Retrieve all or part of a specified configuration datastore
<edit-config>	Edit a configuration datastore by creating, deleting, merging or replacing content
<copy-config>	Copy an entire configuration datastore to another configuration datastore
<delete-config>	Delete a configuration datastore
<lock>	Lock an entire configuration datastore of a device
<unlock>	Release a configuration datastore lock previously obtained with the <lock> operation
<close-session>	Request graceful termination of a NETCONF session
<kill-session>	Force the termination of a NETCONF session

University of Applied Sciences Velika Gorica, Velika Gorica, Croatia

device (of client and server) that may but do not have to be implemented by vendor on the network devices.

In the first part of this paper will be presented an overview of the capabilities of NETCONF protocol. The objective of this paper is to present the results of the interoperability lab where the support for NETCONF capabilities have been tested on router/switch (R/S) devices of major network equipment manufacturers.

II. NETCONF CAPABILITIES

The NETCONF protocol provides a small set of low-level operations to manage device configurations and retrieve device state information. The base protocol provides operations to retrieve, configure, copy, and delete configuration datastores. Additional operations are provided, based on the capabilities advertised by the device [3].

A NETCONF capability is a set of functionality that supplements the base NETCONF specification.

Capabilities augment the base operations of the device, describing both additional operations and the content allowed inside operations. The client can discover the server's capabilities and use any additional operations, parameters, and content defined by those capabilities [3].

The NETCONF capability is identified by a uniform resource identifier (URI) defined in RFC3986. The base capabilities are defined using URNs following the method described in RFC3553 and have the following format:

urn:ietf:params:netconf:capability:{name}:1.x

where {name} is the name of the capability. Capabilities are often referenced using the shorthand *:{name}*[3].

Capabilities are advertised in messages sent by each peer during NETCONF session establishment. When the NETCONF session is opened, each peer (both client and server) must send a <hello> element containing a list of that peer's capabilities. Each peer (client and server) advertises its capabilities by sending them during this initial capabilities exchange. Each peer needs to understand only those capabilities that it might use and must ignore any capability received from the other peer that it does not require or does not understand [3].

Additional capabilities can be defined using the template in RFC6241. Future capability definitions can be published as standards by standards bodies or published as proprietary extensions [3].

Table II. presents the current list of NETCONF capabilities and appropriate RFC documents where they are defined.

III. POSSIBILITIES OF NETCONF PROTOCOL

NETCONF standard offers the following possibilities

TABLE II. CURRENT LIST OF NETCONF CAPABILITIES [4]

Capability	Capability Identifier	Reference
:writable-running	urn:ietf:params:netconf:capability:writable-running:1.0	[RFC6241]
:candidate	urn:ietf:params:netconf:capability:candidate:1.0	[RFC6241]
:confirmed-commit	urn:ietf:params:netconf:capability:confirmed-commit:1.0	[RFC4741]
:confirmed-commit:1.1	urn:ietf:params:netconf:capability:confirmed-commit:1.1	[RFC6241]
:rollback-on-error	urn:ietf:params:netconf:capability:rollback-on-error:1.0	[RFC6241]
:validate	urn:ietf:params:netconf:capability:validate:1.0	[RFC4741]
:validate:1.1	urn:ietf:params:netconf:capability:validate:1.1	[RFC6241]
:startup	urn:ietf:params:netconf:capability:startup:1.0	[RFC6241]
:url	urn:ietf:params:netconf:capability:url:1.0	[RFC6241]
:xpath	urn:ietf:params:netconf:capability:xpath:1.0	[RFC6241]
:notification	urn:ietf:params:netconf:capability:notification:1.0	[RFC5277]
:interleave	urn:ietf:params:netconf:capability:interleave:1.0	[RFC5277]
:partial-lock	urn:ietf:params:netconf:capability:partial-lock:1.0	[RFC5717]
:with-defaults	urn:ietf:params:netconf:capability:with-defaults:1.0	[RFC6243]
:base:1.0	urn:ietf:params:netconf:base:1.0	[RFC4741] [RFC6241]
:base:1.1	urn:ietf:params:netconf:base:1.1	[RFC6241]
:time:1.0	urn:ietf:params:netconf:capability:time:1.0	[RFC7758]
:yang-library	urn:ietf:params:netconf:capability:yang-library:1.0	[RFC7950]
:yang-library:1.1	urn:ietf:params:netconf:capability:yang-library:1.1	[RFC8526]
:with-operational-defaults	urn:ietf:params:netconf:capability:with-operational-defaults:1.0	[RFC8526]

and advantages [5]:

1. NETCONF as a *standard-based solution for configuration management over remote IP access* to R/S device because the structure of commands and data is the same for all vendors.
2. NETCONF as a *standard-based solution for* configuration management because of providing the *automated and programmable network operations* which replaces the process of manual configuration and so significantly decreases the possibility of human error.
3. NETCONF because of making a *clear distinction between configuration and operational data*.
4. Possibility of *partial configuration* of a network device that is offered by NETCONF. NETCONF achieves by using XML language and "XML subtree filtering" function.
5. Possibility of *validation a configuration prior to implementation* that is offered by NETCONF. NETCONF achieves validation of the configuration datastore by using the capability <validate>
6. Possibility of *locking a configuration datastore* offered by NETCONF. NETCONF operations "lock" and "unlock" protect from simultaneous change of configuration datastore from several sources (for example by access of other user with NETCONF session, CLI or SNMP protocol).
7. *"Backup & Restore"* possibility offered by NETCONF. NETCONF achieves by (1) manipulation of different configuration datastores (<running>, <candidate>, <startup> and backup) and (2) by using the "Rollback-on-Error" capability.
8. Possibility of *network-wide configuration* offered by NETCONF. NETCONF achieves by using different configuration datastore (<running>, <candidate>, <startup> and backup) and with operations "lock" and "unlock".

Most of the above listed NETCONF advantages and possibilities are obtained by using NETCONF capabilities.

IV. VENDORS INFORMATION ABOUT NETCONF IMPLEMENTATION

Network devices differ among themselves regarding [5]:

- type of device (e.g. router, switch) and its model;
- size (capacity) of device (e.g. small home router, backbone ISP router);
- vendor (e.g. Cisco, Juniper, …);
- device operating system (IOS, JUNOS, …);
- operating system version.

Two types of devices in computer networks are distinguished [5] [9]:

- end devices: examples are computer, server or IP phone;
- intermediate device: examples are hub, switch, router, firewall or gateway.

All today's computer networks are based on two most widely spread types of intermediate devices: switch and router. The switch performs its basic functions on the OSI model data layer, whereas the router performs its basic functions on the network layer [9].

Support of an R/S (router/switch) device for NETCONF by the vendor depends on the parameters of this device (type of device, device model, type of operating system, operating system version, group of supported functionalities of the operating system) [5].

Because of frequent changes of these parameters it is not possible to determine which R/S network devices support NETCONF and in which software version, and the results change almost daily [5].

An example is the support for NETCONF for Cisco devices. Cisco uses Cisco Feature Navigator application which makes it possible to determine which R/S devices support the NETCONF standard.

Figure 1 shows the example of using Cisco Feature Navigator application in determining the implementation of the NETCONF standard on Cisco devices. Of all the features that are available in the bottom left window, the NETCONF Protocol (6241) has been selected. The bottom left window shows which Platforms (types and models of devices) support NETCONF in IOS XE type of operating system. For the selected Platform of device (router, model ASR1002-HX) the bottom right window shows all software releases that support NETCONF, their requirements for DRAM and Flash memory, as well as the software image.[5]

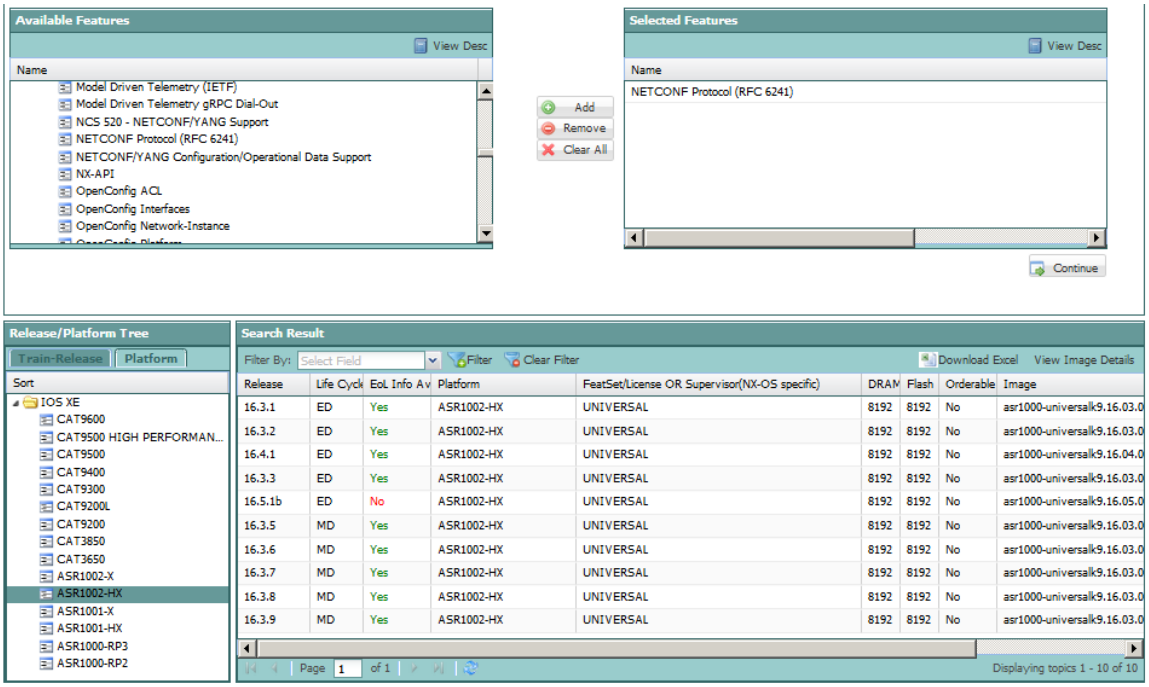

Figure 1 – Cisco Feature Navigator [6]

List of all the devices and their software releases that support NETCONF by the vendor, changes almost daily due to a large number of device models and software versions (e.g. very often a feature is added to certain software versions and devices requested by major users).

Therefore, it is not possible from the vendor's documentation to determine the quantitative implementation of the NETCONF standard on the vendor's devices (e.g. percentage of devices that support NETCONF and which capabilities are implemented).

The implementation of the NETCONF standard is determined in [5][7] through research from the information obtained from the vendors (through interviews) and the users (through interviews and questionnaires). The objective of [4] was to present the results of the interoperability lab where the support for NETCONF capabilities have been tested on R/S devices of major worldwide network equipment manufacturers.

V. INTEROPERABILITY LAB TEST

A. *Selection of R/S vendors for lab test*

For interoperability lab test the vendors with the highest market share of R/S devices have been selected.

Table III. shows the condition on the market of R/S network devices in Croatia based on the research carried out by means of a user questionnaire in [5][7]. It can be seen from the Table III. that there are four vendors most represented by the users in Croatia: Cisco (49.54%), HPE (14.05%), Huawei (9.61%) and Juniper Networks (8.87%) [5][7].

The above information of R/S market share is for Croatia. Similar results are for R/S worldwide market share as shown in IDC report [8]:

- The worldwide Ethernet switch market share for Q1 2019 is: Cisco 53.7%, Huawei 8.9%, Arista Networks 7.5%, HPE 5.3%, Juniper 2.6%, others 22%.
- The worldwide combined SP and enterprise router market share for Q1 2019 is: Cisco 42.4%, Huawei 24.5%, Juniper 10.4%, others 22.7%.

In the lab have been tested R/S devices by Cisco, Juniper Networks and HPE, while Huawei did not make its software available for NETCONF testing. Together, they cover 72.46% of the market of R/S devices in Croatia.

B. *Tested R/S devices and software releases*

Table IV. shows the list of tested virtual R/S devices: device manufacturer (vendor), model of the device

TABLE III. MARKET SHARE OF R/S VENDORS IN CROATIA (USER QUESTIONNAIRE) [5][7]

Vendor	**%**
Cisco	49.54
Hewlett Packard Enterprise (HPE)	14.05
Huawei	9.61
Juniper Networks	8.87
Extreme Networks	4.62
Others	13.31
Total:	100

TABLE IV. LAB TEST: R/S DEVICES AND SOFTWARE VERSIONS [4]

Vendor	Platform	OS	Version	Release date
Cisco	Cisco CRS1000v	Cisco IOS XE	16.9.3	Mar 20 2019
Cisco	Cisco IOS XRv 9000	Cisco IOS XR	6.5.3	Mar 26 2019
Cisco	Cisco Nexus 9000v	Cisco NX-OS	7.0(3)I7(6)	Mar 05 2019
Juniper Networks	Juniper vMX	JUNOS OS	19.1R1.6	Apr 17 2019
HPE	HPE VSR1001	HPE Comware	7.1.064	Jun 13 2017

(platform), type of operating system (OS) used on the device, installed OS version and its date of release.

There has been tested three different platforms by Cisco, each of them with different type of OS (currently Cisco implements three different types of operating system: IOS XE, IOS XR and NX-IOS). [4]

All versions of the software releases that are tested in the lab are for virtual machines and they are released in 2019, except of HPE software that is from 2017 (anyway it was the most recent software for HPE platform).

All OS software used in test lab were downloaded from the vendors' web pages and they were the latest available when the lab testing has been performed (in August 2019).

C. *Test environment*

Figure 2 shows the test environment that consists of NETCONF client and virtual routers and switches.

MG-SOFT NetConf Browser Professional Edition was installed on laptop and has been used in lab test as NETCONF client.

Virtual routers and switches in the test lab had the function of NETCONF servers. EVE-NG and VMware Workstation 15 Player were used to create virtual R/S devices. EVE-NG was installed on virtual server in VMware vSphere virtual environment. VMware Workstation 15 Player was installed on the same laptop as NETCONF client. EVE-NG was used to create virtual routers Cisco IOS XRv 9000 and Cisco CSR 1000v, as well as virtual switch Cisco Nexus 9000v. VMware Workstation 15 Player was used to activate virtual devices Juniper Networks vMX Virtual Router and HPE VSR1000 Virtual Services Router. [4]

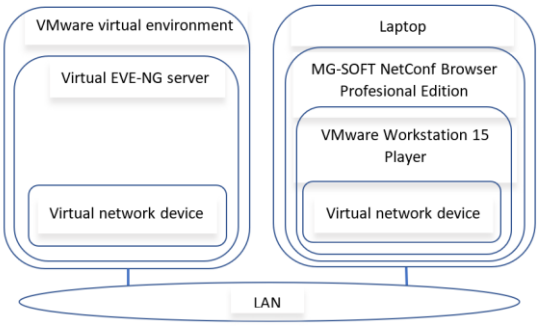

Figure 2 – Test environment [4]

D. *Test results*

Figure 3 shows the result of the lab test: the list of supported capabilities for each tested virtual R/S device.

From the lab test results can be noticed that all three tested Cisco devices support both :base:1.0 and :base:1.1 version of NETCONF protocol, while Juniper Networks and HPE devices support only :base:1.0 version of NETCONF protocol.

Among the tested virtual R/S devices, Cisco CSR 1000v switch supports the highest number of NETCONF capabilities and it is only device that supports :yang-library.

None of tested virtual R/S devices does not support the following NETCONF capabilities: :startup, :partial-lock, :time:1.0, yang-library:1.1, :with-optional-defaults. [4]

VI. CONCLUSION

NETCONF standard defines basic operation set and additional capabilities. Most of the listed NETCONF advantages and possibilities in this article are obtained by using NETCONF capabilities.

In the research [5][7] optional implementation of NETCONF capabilities has been recognized by most of the customers as disadvantage, although some of them see this NETCONF feature as advantage (as it offers

	Cisco Nexus 9000v	Cisco IOS XRv 9000	Cisco CSR 1000v	Juniper Networks vMX	HPE VSR1001
:writable-running	✓	X	✓	X	✓
:candidate	✓	✓	X	✓	X
:confirmed-commit	✓	✓	X	✓	X
:confirmed-commit:1.1	✓	✓	X	X	X
:rollback-on-error	✓	✓	✓	X	✓
:validate	✓	✓	✓	✓	✓
:validate:1.1	✓	✓	✓	X	X
:startup	X	X	X	X	X
:url	X	X	X	✓	X
:xpath	X	X	✓	X	X
:notification	X	✓	✓	X	✓
:interleave	X	✓	✓	X	✓
:partial-lock	X	X	X	X	X
:with-defaults	X	X	✓	X	X
:base:1.0	✓	✓	✓	✓	✓
:base:1.1	✓	✓	✓	X	X
:time:1.0	X	X	X	X	X
:yang-library	X	X	✓	X	X
:yang-library:1.1	X	X	X	X	X
:with-operational-defaults	X	X	X	X	X

Figure 3 – Test results: supported capabilities on R/S devices [4]

flexibility in customer implementation).

It can be very frustrating for the customers in situations where customer could have in their network R/S devices with support of NETCONF standard, but cannot obtain all advantages and possibilities that would be available with all NETCONF capabilities (for example “backup & restore” or network-wide configuration).

The lab test of five different platforms by major worldwide network vendors have shown that:

- all tested R/S devices have support for :base1.0 capability;
- None of tested R/S devices does not support all available capabilities of NETCONF protocol;
- R/S devices from different vendors support different set of NETCONF capabilities;
- different platforms of the same vendor do not support the same set of NETCONF capabilities.

Capabilities are additional functions of devices which may, but do not have to be implemented. On this way NETCONF standard gives a possibility to the vendors not to implement all capabilities, and in this case many of NETCONF advantages could not be used by users.

From the results of lab test it can be seen than some vendors invest more in network programmability approach in order to obtain improved solutions for automated network management.

REFERENCES

[1] Valencic D., Mateljan V.: Implementation of NETCONF Protocol, Mipro 2019.

[2] URL: http://www.ietf.org/rfc.html

[3] IETF: RFC 6241 Network Configuration Protocol (NETCONF), June 2011.

[4] Pupovac V.: Application of capabilities of the NETCONF standard, final work, University of Applied Sciences Velika Gorica, Croatia, (mentor D. Valencic)

[5] Valencic D.: Increasing the availability of network devices by implementing the command configuration standard, doctoral thesis, FFZG, University of Zagreb, Croatia 2020

[6] Cisco Feature Navigator: NETCONF, https://cfn.cloudapps.cisco.com/ITDIT/CFN/jsp/by-feature-technology.jsp (17.08.2019).

[7] Valencic D.: Vendors’ implementation of NETCONF standard on routers and switches, MIPRO 2020.

[8] IDC's Worldwide Ethernet Switch and Router Trackers Show Solid Growth in Both Markets for Q1 2019 https://www.idc.com/getdoc.jsp?containerId=prUS45119319 (17.04.2020).

[9] Valenčić D., Valić B. 2019. Usmjernici i preklopnici: osnovni protokoli i konfiguracija, izdavač Veleučilište Velika Gorica, ISBN 978-953-7716-88-2,

[10] Bajpai, V., Schönwälder, J. 2014. NETCONF Interoperability Lab, Conference paper, 14th IEEE/IFIP Network Operations and Management Symposium (NOMS 2014), At Krakow, Poland.