The Network Configuration Protocol (NETCONF) provides mechanisms to install, manipulate, and delete the configuration of network devices via exchange of XML messages in textual representation. Efficient XML Interchange (EXI) is a W3C-recommended binary representation of XML Information Set, which is more efficient from both CPU and bandwidth utilization perspective. This document defines a capability-based extension to the NETCONF protocol that allows peers to agree to exchange protocol messages using EXI encoding.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 10, 2014.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.
1. Introduction

The NETCONF protocol [RFC6241] is defined in terms of two peers, client and server, exchanging XML messages in an RPC pattern. These messages are encoded as XML text documents, which makes the exchange easily understandable by human operators by simply observing them on the wire. Unfortunately, this feature takes its toll on both computation and network resources, as the representation contains redundant information and verbose names.

Efficient XML Interchange [W3C.REC-exi-20110310] is a W3C Recommendation which defines a more efficient way of encoding XML Information Set than the usual text representation. This is achieved by a combination of reduced verbosity, binary encoding and,
optionally, pruning of non-essential information like comments.

It seems advantageous to allow clients and servers participating on a NETCONF session to sacrifice human readability to increase processing efficiency, especially in environments with high transactional activity and/or limited computing resources.

2. Terminology

This document uses the following terms defined in [RFC6241]:

- capability
- client
- message
- protocol operation
- remote procedure call
- server

3. EXI Capability

3.1. Overview

The :exi capability indicates that the peer supports EXI message encoding and is willing to use it. The capability has no effect on data being handled by the NETCONF protocol, nor does it affect protocol message exchanges.

3.2. Dependencies

EXI-encoded documents are binary data, this capability may only be used when the underlying transport is 8-bit clean and preserves message boundaries in face of arbitrary binary data. Notably this requires use of Chunked Framing mechanism as described in [RFC6242] when used with SSH transport.

3.3. Capability Identifier

The EXI capability is identified by the following capability string:

urn:ietf:params:netconf:capability:exi:1.0

The identifier MAY have a the following parameters:

- compression: This indicates that the sender is willing to perform EXI compression. The parameter MUST contain a positive integral
value, which indicates maximum compression block size which the
sender can process.

schemas: This indicates that the sender can use schema-informed
grammars for EXI encoding. The parameter MUST contain a value,
which has to be one of "builtin" or "base:1.1". The value
"builtin" indicates the ability to use the XML schema built into
the EXI specification. The value of "base:1.1" is a superset of
the "builtin" value and indicates that the sender additionally
supports schema-informed EXI encoding, based on netconf.xsd schema
published in [RFC6241].

Examples:

urn:ietf:params:netconf:capability:exi:1.0?compression=1000000
urn:ietf:params:netconf:capability:exi:1.0?schemas=base:1.1
as=builtin

3.4. New Operations

3.4.1. <start-exi>

Description: The <start-exi> operation requests that the message
encoding be switched to EXI. The operation is invoked by the
client and validated by the server. If the server finds the
parameters acceptable, it will issue a positive response in the
current session encoding. It MUST encode all subsequent messages
using EXI encoding with the supplied parameters. It will also
expect all incoming messages to be EXI-encoded. The client MUST
NOT send any messages to the server between the time is sends this
request and the time it receives a response. Once it receives a
positive reply, it MUST encode all subsequent messages using the
EXI encoding with the parameters supplied in the RPC. If the
operation fails, the session message encoding remains unchanged.

Parameters:

alignment: Requested EXI alignment. If this parameter is not
present, bit-packed is assumed. The following values are
valid:

    bit-packed: Set EXI alignment to bit-packed.

    byte-aligned: Set EXI alignment to byte-aligned.

    pre-compression: Set EXI alignment to pre-compression.

    compressed: Do not specify EXI alignment, but perform EXI
fidelity: Requested EXI fidelity options. If this parameter is not present or empty, all fidelity options are disabled. The following items may be specified:

<comments/>: EXI Preserve.comments
<dtd/>: EXI Preserve.dtd
<lexical-values/>: EXI Preserve.lexicalValues
<pis/>: EXI Preserve.pis
<prefixes/>: EXI Preserve.prefixes

Positive Response: If the device was able to satisfy the request, an <rpc-reply> is sent that contains an <ok> element.

Negative Response: An <rpc-error> element is included in the <rpc-reply> if the request cannot be completed for any reason.

Example:

<rpc message-id="101"
     xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <start-exi>
    <alignment>pre-compression</alignment>
    <fidelity>
      <dtd/>
      <lexical-values/>
    </fidelity>
  </start-exi>
</rpc>

<rpc-reply message-id="101"
            xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>

3.4.2. <stop-exi>
Description: The <stop-exi> operation requests that the message encoding be switched to textual XML. The operation is invoked by the client and validated by the server. If the server is able to switch the encoding to XML, it will issue a positive response in the current session encoding. It MUST encode all subsequent messages using standard XML encoding. It will also expect all incoming messages to be XML-encoded. The client MUST NOT send any messages to the server between the time it sends this request and the time it receives a response. Once it receives a positive reply, it MUST encode all subsequent messages using the standard XML encoding. If the operation fails, the session message encoding remains unchanged. If the session currently uses XML encoding, this RPC is a no-operation and SHOULD NOT fail.

Positive Response: If the device was able to satisfy the request, an <rpc-reply> is sent that contains an <ok> element.

Negative Response: An <rpc-error> element is included in the <rpc-reply> if the request cannot be completed for any reason.

Example:

```xml
<rpc message-id="101"
     xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <stop-exi/>
</rpc>

<rpc-reply message-id="101"
            xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

4. YANG module for <start-exi> and <stop-exi> Operations

The following YANG module defines the new operations introduced in this document. The YANG language is defined in [RFC6020]. Every NETCONF speaker that supports the :exi capability MUST implement this YANG module.

TBD.

5. IANA considerations

This document registers the following capability identifier URN in the ‘Network Configuration Protocol (NETCONF) Capability URNs’ registry: urn:ietf:params:netconf:capability:exi:1.0

6. Security Considerations

The compression option present in EXI specification may increase CPU and memory requirements for encoding the response. Devices should ensure this overhead is acceptable before agreeing to using EXI encoding, such that no operational risks are introduced.
7. Acknowledgements

The author would like to thank Anton Tkacik, Miroslav Miklus and Stefan Kobza for their contributions to this document.

8. Normative References


Author’s Address

Robert Varga
Pantheon Technologies SRO
Mlynske Nivy 56
Bratislava 821 05
Slovakia

Email: robert.varga@pantheon.sk