Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 18, 2018

M. Jethanandani
Cisco Systems, Inc

July 17, 2017

Accounting in NETCONF and RESTCONF
draft-mahesh-netconf-accounting-03

Abstract

This document defines an accounting record for NETCONF and RESTCONF.

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 RFC 2119 [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 18, 2018.

Copyright Notice

Table of Contents

1. Introduction

NETCONF [RFC6241] and RESTCONF [RFC8040] protocol operations are authenticated and authorized as part of the Authentication, Authorization and Accounting (AAA) framework. An accounting record is generated as part of the same framework for each of these operations to satisfy the accounting part of AAA, but there has been no effort to define such a record. Having an accounting record that is consistent across vendors allows for the operator to compare operations across devices from different vendors. This document defines such a record and a corresponding YANG data model (ietf-netconf-am.yang).

The rest of this document will use NETCONF to imply both NETCONF and RESTCONF, but where applicable will call out each protocol specifically.

1.1. Compatability with remote AAA servers

This document does not cover how the server interacts with remote AAA servers and any interaction is out of scope of this document. A particular implementation can make the records available as part of <get> request, send a notification every time a accounting record is

generated or use any existing protocol to update the remote AAA server.

1.2. Terminology

The following terms are defined in NETCONF [RFC6241] and are not redefined here:

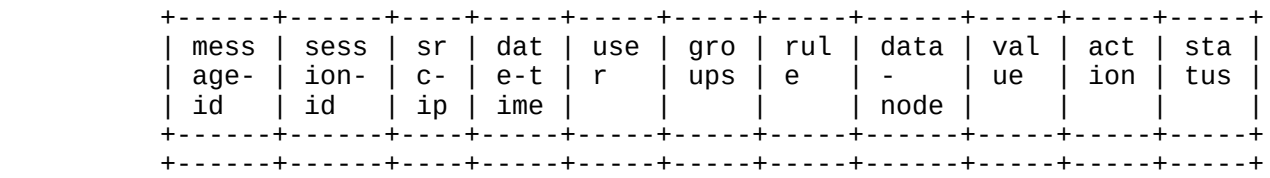
- client
- <get>
- notification
- server
- session
- user

And the following terms are defined in NACM [RFC6536] and are not redefined here.

- data-node
- action
- rule

2. Accounting Record

An accounting record for NETCONF consists of the following fields. Note, there is no accounting record for reading or notification of an accounting record.



where:

message-id: This is the id within a given NETCONF session assigned to each RPC. RESTCONF has no concept of a session, so this field would be left blank.

session-id: The session-id in case of NETCONF and would be blank in case of RESTCONF. If the accounting record needs to be fragmented for any reason, it is suggested that this field not be repeated in subsequent packets. Instead a combination of start and end record marker, and the message-id should be used to reassemble fragmented records.

src-ip: The source IP address that was used to request the operation. If the accounting record needs to be fragmented for any reason, it is suggested that this field not be repeated in subsequent packets. Instead a combination of start and end record marker, and the message-id should be used to reassemble fragmented records.

date-time: The date and time when the operation was performed (UTC Timezone). If the accounting record needs to be fragmented for any reason, it is suggested that this field not be repeated in subsequent packets. Instead a combination of start and end record marker, and the message-id should be used to reassemble fragmented records.

user: The NETCONF user that requesed this operation. If the accounting record needs to be fragmented for any reason, it is suggested that this field not be repeated in subsequent packets. Instead a combination of start and end record marker, and the message-id should be used to reassemble fragmented records.

groups: The group the user belongs to. If the accounting record needs to be fragmented for any reason, it is suggested that this field not be repeated in subsequent packets. Instead a combination of start and end record marker, and the message-id should be used to reassemble fragmented records.

data-node: The data-node in the NACM [RFC6536] rule on which the operations is being performed

value: The value that was set for any of the attributes in the request

action: The action in the NACM [RFC6536] rule

rule: The rule in the NACM [RFC6536] that was matched to authorize the action.

status: Whether the operations was permitted or denied.

3. Data Model Definitions

The model uses the NACM extension statement of default-deny-all to protect accounting records. Explicit rules have to be defined to be enable access to the accounting records.

3.1. Data Organization

The following diagram highlights the contents and structure of the Accounting YANG module. For information on annotations, please refer to YANG Tree Diagrams [I-D.ietf-netmod-yang-tree-diagrams].

```
module: ietf-netconf-am
    +--rw nam
       +--rw enable-nam?          boolean
       +--ro accounting-record* [session-id message-id]
          +--ro session-id    nc:session-id-type
          +--ro message-id    uint32
          +--ro date-time     yang:date-and-time
          +--ro src-ip        inet:ip-address
          +--ro group         nacm:group-name-type
          +--ro user?         nacm:user-name-type
          +--ro rule?         string
          +--ro data-node     nacm:node-instance-identifier
          +--ro value?
          +--ro action        nacm:access-operations-type
          +--ro status?       nacm:action-type
```

3.2. YANG Module

The following YANG module specifies the normative NETCONF content that MUST be supported by the server.

The "ietf-netconf-am" YANG module imports typedefs from YANG-TYPES [RFC6991], from NETCONF [RFC6241] and from NACM [RFC6536].

<CODE BEGINS> file "ietf-netconf-am@2017-07-16.yang"

```
module ietf-netconf-am {
    yang-version 1.1;
    namespace "urn:ietf:params:xml:ns:yang:ietf-netconf-am";
    prefix "nam";

    import ietf-inet-types {
        prefix inet;
    }

    import ietf-yang-types {
```

```
        prefix yang;
    }

    import ietf-netconf {
        prefix nc;
    }

    import ietf-netconf-acm {
        prefix nacm;
    }

    organization
        "IETF NETCONF (Network Configuration) Working Group";

    contact
        "WG Web:   <http://tools.ietf.org/wg/netconf/>
         WG List:  <mailto:netconf@ietf.org>

         WG Chair: Mehmet Ersue
                   <mailto:mehmet.ersue@nsn.com>

         WG Chair: Mahesh Jethanandani
                   <mailto:mjethanandani@gmail.com>

         Editor:   Mahesh Jethanandani
                   <mailto:mjethanandani@gmail.com>";


    description
        "This module defines an accounting record for NETCONF operations
         performed on the server. If these operations are authorized
         using rules defined by NACM [RFC6536], then that information is
         also captured by this module.

         Copyright (c) 2014 IETF Trust and the persons identified as
         authors of the code. All rights reserved.

         Redistribution and use in source and binary forms, with or
         without modification, is permitted pursuant to, and subject
         to the license terms contained in, the Simplified BSD
         License set forth in Section 4.c of the IETF Trust's
         Legal Provisions Relating to IETF Documents
         (http://trustee.ietf.org/license-info).

         This version of this YANG module is part of RFC XXXX; see
         the RFC itself for full legal notices.";

    revision "2017-07-16" {
```

```
         description
             "Initial version";
         reference
             "RFC XXXX: NETCONF and RESTCONF Accounting";
    }

    /*
     * Data definition statements.
     */

    container nam {
        nacm:default-deny-all;

        description
            "Parameters for NETCONF Accounting Model.";

        leaf enable-nam {
                type boolean;
            default true;
            description
                "Enable or disable generation of NETCONF
                 accounting records. If 'true', accounting
                 records will be generated. If set to 'false'
                 no accounting records will be generated.";
        }

        list accounting-record {
            key "session-id message-id";
            config false;
            description
                "A list of accounting records generated by the server";

            leaf session-id {
                type nc:session-id-type;
                description
                    "If this operation happened over NETCONF, this
                     field captures the NETCONF session-id. In case
                     of RESTCONF this field can be left blank.";
            }

            leaf message-id {
                type uint32;
                description
                    "Id that is assigned to each RPC within a given
                     NETCONF session. Should be blank in case of
                     RESTCONF.";
            }
```

```
            leaf date-time {
                type yang:date-and-time;
                mandatory true;
                description
                    "The date and time when the operation was
                     requested.";
            }

            leaf src-ip {
                type inet:ip-address;
                mandatory true;
                description
                    "The source IP address where the request was made
                     from.";
            }

            leaf group {
                type nacm:group-name-type;
                mandatory true;
                description
                    "The name of the group that the user who requested
                     the operation belongs to.";
            }

            leaf user {
                type nacm:user-name-type;
                description
                    "The user within the group that is requesting this
                     operation.";
            }

            leaf rule {
                type string {
                    length "1..max";
                }
                description
                    "The name assigned to the rule that was used to
                     authorize the action, if authorization was
                     enabled.";
            }

            leaf data-node {
                type nacm:node-instance-identifier;
                mandatory true;
                description
                    "Data Node Instance Identifier associated with the
                     data node that the request is being made on.
```

```
                    Instance identifiers start with the top-level
                    data node, and a complete identifier is required
                    for this value.";
            }

            anydata value {
                description
                    "An optional field, it contains the value of any
                     of the attribute that form the record.

                     It could be as simple as the filter value
                     'http' specified that the user requested as part
                     of the authorization request such as in this
                     example:

                     <filter>
                         <name>http</name>
                     </filter>

                     or it could be value being set for a ssh port
                     in this example:

                     <ssh>
                         <port>2022</port>
                     </ssh>";
            }

            leaf action {
                type nacm:access-operations-type;
                mandatory true;
                description
                    "The type of NETCONF operation being requested.";
            }

            leaf status {
                type nacm:action-type;
                description
                    "Action taken by the server when the above
                     mentioned rule matched, if authorization was
                     enable.";
            }
        }
    }
}
```

<CODE ENDS>

4. IANA Considerations

This document makes two requests of IANA.

The first request is to register one URI in "The IETF XML Registry". Following the format in The IETF XML Registry [RFC3688], the following needs to be registered.

URI: urn:ietf:params:xml:ns:yang:ietf-netconf-am

Registrant Contact: The IESG

XML: N/A, the requested URI is an XML namespace

The second request is to register one module in the "YANG Module Names" registry. Following the format in YANG [RFC7950], the following needs to be registered.

Name: ietf-netconf-am

Namespace: urn:ietf:params:xml:ns:yang:ietf-netconf-am

Prefix: nam

Reference: RFC XXXX

Note to RFC Editor - Please replace XXXX here and in the rest of the draft with the RFC id assigned to this draft.

5. Security Considerations

The YANG module defined in this document is designed to be accessed via network management protocol such as NETCONF [RFC6241] or RESTCONF [RFC8040]. The lowest NETCONF layer is the secure transport layer, and the mandatory-to-implement secure transport is Secure Shell (SSH) [RFC6242]. The lowest RESTCONF layers is HTTPS, and the mandatory-to-implement secure transport is TLS [RFC5246].

The NETCONF Access Control Model (NACM) [RFC6536] provides the means to restrict access for particular NETCONF or RESTCONF users to a pre-configured subset of all available NETCONF or RESTCONF protocol operations and content.

Most of the data nodes defined in this YANG module are readonly, i.e. config false, and are therefore not vulnerable to manipulation in network environments. However, they might contain data that might be sensitive and should be protected with the right NACM [RFC6536] rules.

6. Acknowledgements

7. References

7.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

[RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, DOI 10.17487/RFC3688, January 2004, <http://www.rfc-editor.org/info/rfc3688>.

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, DOI 10.17487/RFC5246, August 2008, <http://www.rfc-editor.org/info/rfc5246>.

[RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <http://www.rfc-editor.org/info/rfc6241>.

[RFC6242] Wasserman, M., "Using the NETCONF Protocol over Secure Shell (SSH)", RFC 6242, DOI 10.17487/RFC6242, June 2011, <http://www.rfc-editor.org/info/rfc6242>.

[RFC6536] Bierman, A. and M. Bjorklund, "Network Configuration Protocol (NETCONF) Access Control Model", RFC 6536, DOI 10.17487/RFC6536, March 2012, <http://www.rfc-editor.org/info/rfc6536>.

[RFC6991] Schoenwaelder, J., Ed., "Common YANG Data Types", RFC 6991, DOI 10.17487/RFC6991, July 2013, <http://www.rfc-editor.org/info/rfc6991>.

[RFC7950] Bjorklund, M., Ed., "The YANG 1.1 Data Modeling Language", RFC 7950, DOI 10.17487/RFC7950, August 2016, <http://www.rfc-editor.org/info/rfc7950>.

[RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017, <http://www.rfc-editor.org/info/rfc8040>.

7.2. Informative References

[I-D.ietf-netmod-yang-tree-diagrams]
Bjorklund, M. and L. Berger, "YANG Tree Diagrams", draft-ietf-netmod-yang-tree-diagrams-01 (work in progress), June 2017.

Appendix A. Accounting model examples

A.1. <edit-config> Example for Accounting

This example demonstrates how the configuration could be updated to enable accounting. The attribute in the model that enables accounting is enable-nam.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
     message-id="4">
  <edit-config xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
    <target>
      <candidate/>
    </target>
    <test-option>test-then-set</test-option>
    <error-option>rollback-on-error</error-option>
    <config>
      <nam xmlns="urn:ietf:params:xml:ns:yang:ietf-netconf-am">
        <enable-nam>true</enable-nam>
      </nam>
    </config>
  </edit-config>
</rpc>
```

A.2. <get> Example for Accounting

This example demonsrates what a <get> request and response might look like.

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
     message-id="1">
  <get>
    <filter>
      <nam xmlns="urn:ietf:params:xml:ns:yang:ietf-netconf-am">
        <accounting-record/>
      </nam>
    </filter>
  </get>
</rpc>

<rpc-reply message-id="1"
           xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <data>
    <nam xmlns="urn:ietf:params:xml:ns:yang:ietf-netconf-am">
      <accounting-record>
        <session-id>101</session-id>
        <message-id>100</message-id>
        <date-time>2017-06-19T16:39:57-08:00</date-time>
        <src-ip>172.20.39.46</src-ip>
        <group>netconf-group</group>
        <user>netconf</user>
        <path xmlns:acme="http://example.com/ns/itf">
          /acme:interfaces/acme:interface
        </path>
        <value>
          <name>GigabitEthernet0/0/0/0</name>
          <admin-state>UP</admin-state>
        </value>
        <action>read</action>
        <rule>51</rule>
        <status>permit</status>
      </accounting-record>
    </nam>
  </data>
</rpc-reply>
```

A.3. NACM rule for Accounting

This example demonstrates how NACM could be configured to permit access to accounting records. Note, the model has default-deny-all set to prevent access to accounting records by default, and expects explicit rules to be configured to permit access.

```
   <nacm xmlns="urn:ietf:params:xml:ns:yang:ietf-netconf-acm">
     <rule-list>
       <name>root</name>
       <group>root</group>

       <rule>
         <name>allow-nam</name>
         <path xmlns:n="urn:ietf:params:xml:ns:yang:ietf-netconf-am">
           /n:nam
         </path>
         <access-operations>read</access-operations>
         <action>permit</action>
         <comment>
           Allow the root group read access to the /nam data.
         </comment>
       </rule>
     </rule-list>
   </nacm>
```

Author’s Address

Mahesh Jethanandani
Cisco Systems, Inc
170 West Tasman Drive
San Jose, CA 95070
USA

Email: mjethanandani@gmail.com