Opsawg
Internet-Draft
Intended status: Standards Track
Expires: October 10, 2021

B. Wu, Ed.
G. Zheng
M. Wang, Ed.
Huawei

April 8, 2021

A YANG Module for TACACS+
draft-ietf-opsawg-tacacs-yang-10

Abstract

This document defines a Terminal Access Controller Access-Control System Plus (TACACS+) client YANG module, that augments the System Management data model, defined in RFC 7317, to allow devices to make use of TACACS+ servers for centralized Authentication, Authorization and Accounting (AAA).

The YANG module in this document conforms to the Network Management Datastore Architecture (NMDA) defined in RFC 8342.

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 https://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 October 10, 2021.

Copyright Notice

Copyright (c) 2021 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (https://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect

to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 2
2. Conventions used in this document 3
2.1. Tree Diagrams 3
3. Design of the TACACS+ Data Model 3
4. TACACS+ Client Module 5
5. Security Considerations 12
6. IANA Considerations 13
7. Acknowledgments 13
8. References 13
8.1. Normative References 13
8.2. Informative References 15
Appendix A. Example TACACS+ Authentication Configuration 15
Authors' Addresses 15

1. Introduction

This document defines a YANG module that augments the System Management data model defined in the [RFC7317] to support the configuration and management of TACACS+ clients.

TACACS+ [RFC8907] provides device administration for routers, network access servers and other networked devices via one or more centralized servers.

The System Management Model [RFC7317] defines separate functionality to support local and RADIUS authentication:

- o User Authentication Model: Defines a list of usernames with associated passwords and a configuration leaf to decide the order in which local or RADIUS authentication is used.

- o RADIUS Client Model: Defines a list of RADIUS servers used by a device for centralized user authentication.

The System Management Model is augmented with the TACACS+ YANG module defined in this document to allow the use of TACACS+ servers as an alternative to RADIUS servers or local user configuration.

The YANG module can be used with network management protocols such as NETCONF[RFC6241].

The YANG module in this document conforms to the Network Management Datastore Architecture (NMDA) defined in [RFC8342].

2. Conventions used in this document

The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP14, [RFC2119], [RFC8174] when, and only when, they appear in all capitals, as shown here.

The following terms are defined in [RFC6241] and are used in this specification:

- o configuration data
- o state data

The following terms are defined in [RFC7950] and are used in this specification:

- o augment
- o data model
- o data node

The terminology for describing YANG data models is found in [RFC7950].

2.1. Tree Diagrams

Tree diagrams used in this document follow the notation defined in [RFC8340].

3. Design of the TACACS+ Data Model

This module is used to configure TACACS+ client on a device to support deployment scenarios with centralized authentication, authorization, and accounting servers. Authentication is used to validate a user's username and password, authorization allows the user to access and execute commands at various command levels assigned to the user, and accounting keeps track of the activity of a user who has accessed the device.

The ietf-system-tacacs-plus module augments the "/sys:system" path defined in the ietf-system module with the contents of the"tacacs-plus" grouping. Therefore, a device can use local, RADIUS, or

TACACS+ to validate users who attempt to access the router by several mechanisms, e.g., a command line interface or a web-based user interface.

The "server" list is directly under the "tacacs-plus" container, which holds a list of TACACS+ servers and uses server-type to distinguish between Authentication, Authorization and Accounting (AAA). The list of servers is for redundancy.

Most of the parameters in the "server" list are taken directly from the TACACS+ protocol [RFC8907], and some are derived from the various implementations by network equipment manufacturers. For example, when there are multiple interfaces connected to the TACACS+ client or server, the source address of outgoing TACACS+ packets could be specified, or the source address could be specified through the interface IP address setting, or derived from the outbound interface from the local Forwarding Information Base (FIB). For the TACACS+ server located in a Virtual Private Network (VPN), a VPN Routing and Forwarding (VRF) instance needs to be specified.

The "statistics" container under the "server list" is a collection of read-only counters for sent and received messages from a configured server.

The YANG module for TACACS+ client has the following structure:

```
module: ietf-system-tacacs-plus
  augment /sys:system:
    +--rw tacacs-plus
       +--rw server* [name]
          +--rw name                      string
          +--rw server-type?              tacacs-plus-server-type
          +--rw address                   inet:host
          +--rw port?                     inet:port-number
          +--rw (encryption)
          |  +--:(shared-secret)
          |     +--rw shared-secret?      string
          +--rw (source-type)?
          |  +--:(source-ip)
          |  |  +--rw source-ip?          inet:ip-address
          |  +--:(source-interface)
          |     +--rw source-interface?   if:interface-ref
          +--rw vrf-instance?
          |       -> /ni:network-instances/network-instance/name
          +--rw single-connection?        boolean
          +--rw timeout?                  uint16
          +--ro statistics
             +--ro connection-opens?      yang:counter64
             +--ro connection-closes?     yang:counter64
             +--ro connection-aborts?     yang:counter64
             +--ro connection-failures?   yang:counter64
             +--ro connection-timeouts?   yang:counter64
             +--ro messages-sent?         yang:counter64
             +--ro messages-received?     yang:counter64
             +--ro errors-received?       yang:counter64
             +--ro sessions?              yang:counter64
```

4. TACACS+ Client Module

This YANG module imports typedefs from [RFC6991]. This module also uses the interface typedef from [RFC8343], the leafref to VRF instance from [RFC8529], and the "default-deny-all" extension statement from [RFC8341].

<CODE BEGINS> file "ietf-system-tacacs-plus@2021-04-08.yang"

```
module ietf-system-tacacs-plus {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-system-tacacs-plus";
  prefix sys-tcs-plus;

  import ietf-inet-types {
    prefix inet;
    reference
```

```
      "RFC 6991: Common YANG Data Types";
  }
  import ietf-yang-types {
    prefix yang;
    reference
      "RFC 6991: Common YANG Data Types";
  }
  import ietf-network-instance {
    prefix ni;
    reference
      "RFC 8529: YANG Data Model for Network Instances";
  }
  import ietf-interfaces {
    prefix if;
    reference
      "RFC 8343: A YANG Data Model for Interface Management";
  }
  import ietf-system {
    prefix sys;
    reference
      "RFC 7317: A YANG Data Model for System Management";
  }
  import ietf-netconf-acm {
    prefix nacm;
    reference
      "RFC 8341: Network Configuration Access Control Model";
  }

  organization
    "IETF Opsawg (Operations and Management Area Working Group)";
  contact
    "WG Web:   <http://tools.ietf.org/wg/opsawg/>
     WG List:  <mailto:opsawg@ietf.org>

     Editor: Bo Wu <lana.wubo@huawei.com>
     Editor: Guangying Zheng <zhengguangying@huawei.com>";
  description
    "This module provides configuration of TACACS+ client.

     Copyright (c) 2021 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.";

  // RFC Ed.: update the date below with the date of RFC
  // publication and remove this note.
  // RFC Ed.: replace XXXX with actual RFC number and remove
  // this note.

  revision 2021-04-08 {
    description
      "Initial revision.";
    reference
      "RFC XXXX: A YANG Module for TACACS+";
  }

  typedef tacacs-plus-server-type {
    type bits {
      bit authentication {
        description
          "Indicates that the TACACS server is providing authentication
           services.";
      }
      bit authorization {
        description
          "Indicates that the TACACS server is providing authorization
           services.";
      }
      bit accounting {
        description
          "Indicates that the TACACS server is providing accounting
           services.";
      }
    }
    description
      "tacacs-plus-server-type can be set to
       authentication/authorization/accounting
       or any combination of the three types.";
  }

  identity tacacs-plus {
    base sys:authentication-method;
    description
      "Indicates AAA operation using TACACS+.";
    reference
      "RFC 8907: The TACACS+ Protocol";
  }

  grouping statistics {
```

```
    description
      "Grouping for TACACS+ statistics attributes";
    container statistics {
      config false;
      description
        "A collection of server-related statistics objects";
      leaf connection-opens {
        type yang:counter64;
        description
          "Number of new connection requests sent to the server, e.g.,
           socket open";
      }
      leaf connection-closes {
        type yang:counter64;
        description
          "Number of connection close requests sent to the server, e.g.,
           socket close";
      }
      leaf connection-aborts {
        type yang:counter64;
        description
          "Number of aborted connections to the server. These do
           not include connections that are close gracefully.";
      }
      leaf connection-failures {
        type yang:counter64;
        description
          "Number of connection failures to the server";
      }
      leaf connection-timeouts {
        type yang:counter64;
        description
          "Number of connection timeouts to the server";
      }
      leaf messages-sent {
        type yang:counter64;
        description
          "Number of messages sent to the server";
      }
      leaf messages-received {
        type yang:counter64;
        description
          "Number of messages received from the server";
      }
      leaf errors-received {
        type yang:counter64;
        description
          "Number of error messages received from the server";
```

```
      }
      leaf sessions {
        type yang:counter64;
        description
          "Number of TACACS+ sessions completed with the server.
           If the Single Connection Mode was NOT enabled, the number of
           sessions is the same as the number of 'connection-closes'.
           If the Mode was enabled, a single TCP connection may contain
           multiple TACACS+ sessions.";
      }
    }
  }

  grouping tacacs-plus {
    description
      "Grouping for TACACS+ attributes";
    container tacacs-plus {
      must "not(derived-from-or-self(../sys:authentication"
         + "/sys:user-authentication-order, 'tacacs-plus'))"
         + " or bit-is-set(server/server-type,'authentication')" {
        error-message "When 'tacacs-plus' is used as a system"
                    + " authentication method, a TACACS+ authentication"
                    + " server must be configured.";
        description
          "When 'tacacs-plus' is used as an authentication method,
           a TACACS+ server must be configured.";
      }
      description
        "Container for TACACS+ configurations and operations.";
      list server {
        key "name";
        ordered-by user;
        description
          "List of TACACS+ servers used by the device.";
        leaf name {
          type string;
          description
            "An arbitrary name for the TACACS+ server.";
        }
        leaf server-type {
          type tacacs-plus-server-type;
          description
            "Server type: authentication/authorization/accounting and
             various combinations.";
        }
        leaf address {
          type inet:host;
          mandatory true;
```

```
          description
            "The address of the TACACS+ server.";
        }
        leaf port {
          type inet:port-number;
          default "49";
          description
            "The port number of TACACS+ Server port.";
        }
        choice encryption {
          mandatory true;
          description
            "Encryption mechanism between TACACS+ client and server.";
          case shared-secret {
            leaf shared-secret {
              type string {
                length "16..max";
              }
              nacm:default-deny-all;
              description
                "The shared secret, which is known to both the
                 TACACS+ client and server. TACACS+ server
                 administrators should configure shared secret of
                 minimum 16 characters length.
                 It is highly recommended that this shared secret is
                 at least 32 characters long and sufficiently complex
                 with a mix of different character types
                 i.e. upper case, lower case, numeric, punctuation.";
              reference
                "RFC 8907: The TACACS+ Protocol";
            }
          }
        }
        choice source-type {
          description
            "The source address type for outbound TACACS+ packets.";
          case source-ip {
            leaf source-ip {
              type inet:ip-address;
              description
                "Specifies source IP address for TACACS+ outbound
                 packets.";
            }
          }
          case source-interface {
            leaf source-interface {
              type if:interface-ref;
              description
```

```
                "Specifies the interface from which the IP address is
                 derived for use as the source for the outbound TACACS+
                 packet";
            }
          }
        }
        leaf vrf-instance {
          type leafref {
            path "/ni:network-instances/ni:network-instance/ni:name";
          }
          description
            "Specifies the VPN Routing and Forwarding (VRF) instance to
             use to communicate with the TACACS+ server.";
          reference
            "RFC 8529: YANG Data Model for Network Instances";
        }
        leaf single-connection {
          type boolean;
          default "false";
          description
            "Whether the single connection mode is enabled for the
             server. By default, the single connection mode is
             disabled.";
        }
        leaf timeout {
          type uint16 {
            range "1..300";
          }
          units "seconds";
          default "5";
          description
            "The number of seconds the device will wait for a
             response from each TACACS+ server before trying with a
             different server.";
        }
        uses statistics;
      }
    }
  }

  augment "/sys:system" {
    description
      "Augment the system model with the tacacs-plus model";
    uses tacacs-plus;
  }
}
```

<CODE ENDS>

5. Security Considerations

The YANG module defined in this document is designed to be accessed via network management protocols 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 layer is HTTPS, and the mandatory-to-implement secure transport is TLS [RFC8446].

The NETCONF access control model [RFC8341] provides the means to restrict access for particular NETCONF or RESTCONF users to a preconfigured subset of all available NETCONF or RESTCONF protocol operations and content.

There are a number of data nodes defined in this YANG module that are writable/creatable/deletable (i.e., config true, which is the default). These data nodes may be considered sensitive or vulnerable in some network environments. Write operations (e.g., edit-config) to these data nodes without proper protection can have a negative effect on network operations. These are the subtrees and data nodes and their sensitivity/vulnerability:

/system/tacacsplus/server: This list contains the data nodes used to control the TACACS+ servers used by the device. Unauthorized access to this list could cause a complete control over the device by pointing to a compromised TACACS+ server.

/system/tacacsplus/server/shared-secret: This leaf controls the key known to both the TACACS+ client and server. Unauthorized access to this leaf could make the device vulnerable to attacks, therefore has been restricted using the "default-deny-all" access control defined in [RFC8341]. When setting, it is highly recommended that the leaf is at least 32 characters long and sufficiently complex with a mix of different character types i.e. upper case, lower case, numeric, punctuation.

This document describes the use of TACACS+ for purposes of authentication, authorization and accounting, it is vulnerable to all of the threats that are present in TACACS+ applications. For a discussion of such threats, see Section 9 of the TACACS+ Protocol [RFC8907].

6. IANA Considerations

This document registers a URI in the IETF XML registry [RFC3688]. Following the format in [RFC3688], the following registration is requested to be made:

```
URI: urn:ietf:params:xml:ns:yang:ietf-system-tacacs-plus
Registrant Contact: The IESG.
XML: N/A, the requested URI is an XML namespace.
```

This document registers a YANG module in the YANG Module Names registry [RFC7950].

```
Name: ietf-system-tacacs-plus
Namespace: urn:ietf:params:xml:ns:yang:ietf-system-tacacs-plus
Prefix: sys-tcs-plus
Reference: RFC XXXX (RFC Ed.: replace XXXX with actual
RFC number and remove this note.)
```

7. Acknowledgments

The authors wish to thank Alex Campbell, John Heasley, Ebben Aries, Alan DeKok, Joe Clarke, Joe Clarke, Tom Petch, Robert Wilton, and many others for their helpful comments and suggestions.

8. References

8.1. 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, <https://www.rfc-editor.org/info/rfc2119>.

[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, <https://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, <https://www.rfc-editor.org/info/rfc6242>.

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

[RFC7317] Bierman, A. and M. Bjorklund, "A YANG Data Model for System Management", RFC 7317, DOI 10.17487/RFC7317, August 2014, <https://www.rfc-editor.org/info/rfc7317>.

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

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

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8340] Bjorklund, M. and L. Berger, Ed., "YANG Tree Diagrams", BCP 215, RFC 8340, DOI 10.17487/RFC8340, March 2018, <https://www.rfc-editor.org/info/rfc8340>.

[RFC8341] Bierman, A. and M. Bjorklund, "Network Configuration Access Control Model", STD 91, RFC 8341, DOI 10.17487/RFC8341, March 2018, <https://www.rfc-editor.org/info/rfc8341>.

[RFC8342] Bjorklund, M., Schoenwaelder, J., Shafer, P., Watsen, K., and R. Wilton, "Network Management Datastore Architecture (NMDA)", RFC 8342, DOI 10.17487/RFC8342, March 2018, <https://www.rfc-editor.org/info/rfc8342>.

[RFC8343] Bjorklund, M., "A YANG Data Model for Interface Management", RFC 8343, DOI 10.17487/RFC8343, March 2018, <https://www.rfc-editor.org/info/rfc8343>.

[RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <https://www.rfc-editor.org/info/rfc8446>.

[RFC8529] Berger, L., Hopps, C., Lindem, A., Bogdanovic, D., and X. Liu, "YANG Data Model for Network Instances", RFC 8529, DOI 10.17487/RFC8529, March 2019, <https://www.rfc-editor.org/info/rfc8529>.

[RFC8907] Dahm, T., Ota, A., Medway Gash, D., Carrel, D., and L. Grant, "The Terminal Access Controller Access-Control System Plus (TACACS+) Protocol", RFC 8907, DOI 10.17487/RFC8907, September 2020, <https://www.rfc-editor.org/info/rfc8907>.

8.2. Informative References

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

Appendix A. Example TACACS+ Authentication Configuration

The following shows an example where a TACACS+ authentication server instance is configured.

```
{
  "ietf-system:system": {
    "authentication": {
      "user-authentication-order": [tacacs-plus, local-users]
    }
    "tacacs-plus": {
      "server": [
        {
          "name": "tac_plus1",
          "server-type": "authentication",
          "address": "192.0.2.2",
          "shared-secret": "QaEfThUkO198010075460923+h3TbE8n",
          "source-ip": "192.0.2.12",
          "timeout": "10"
        }
      ]
    }
  }
}
```

Authors' Addresses

Bo Wu (editor)
Huawei Technologies, Co.,
Ltd
101 Software Avenue, Yuhua District
Nanjing, Jiangsu 210012
China

Email: lana.wubo@huawei.com

Guangying Zheng
Huawei Technologies, Co.,
Ltd
101 Software Avenue, Yuhua District
Nanjing, Jiangsu 210012
China

Email: zhengguangying@huawei.com

Michael Wang (editor)
Huawei Technologies, Co.,
Ltd
101 Software Avenue, Yuhua District
Nanjing 210012
China

Email: wangzitao@huawei.com