RFC 9105
A YANG Data Model for Terminal Access Controller Access-Control System Plus (TACACS+)

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). Though being a standard module, this module does not endorse the security mechanisms of the TACACS+ protocol (RFC 8907), and TACACS+ MUST be used within a secure deployment.

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

Status of This Memo
This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at https://www.rfc-editor.org/info/rfc9105.

Copyright Notice
Copyright (c) 2021 IETF Trust and the persons identified as the document authors. All rights reserved.
This document defines a YANG module that augments the System Management data model defined in [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 data model [RFC7317] defines separate functionality to support local and RADIUS authentication:

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

The System Management data 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.

The YANG module can be used with network management protocols such as the Network Configuration Protocol (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 key words "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 BCP 14 [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:

- configuration data
- state data

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

- augment
- data model
- data node

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

2.1. Tree Diagrams

The tree diagram used in this document follows the notation defined in [RFC8340].

3. Design of the TACACS+ Data Model

This module is used to configure a 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 privilege 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+ authentication 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, which is directly under the "tacacs-plus" container, holds a list of TACACS+ servers and uses server-type to distinguish between Authentication, Authorization, and Accounting (AAA) services. 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:

```yang
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 (security)
          | +--:(obfuscation)
          | | +--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].
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://datatracker.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."

    The key words '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 BCP 14 (RFC 2119) (RFC 8174) when, and only when, they appear in all capitals, as shown here.

    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 (https://trustee.ietf.org/license-info).

This version of this YANG module is part of RFC 9105; see the RFC itself for full legal notices.

revision 2021-08-05 {
  description
    "Initial revision.";
  reference
    "RFC 9105: A YANG Data Model for Terminal Access Controller Access-Control System Plus (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 closed 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')"
  + " 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;
    mandatory true;
    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 security {
    mandatory true;
    description
    "Security mechanism between TACACS+ client and server. This is modeled as a YANG 'choice' so that it can be augmented by a YANG module in a backwards-compatible manner.";
    case obfuscation {
      leaf shared-secret {
        type string {
          length "1..max";
        }
        nacm:default-deny-all;
        description
        "The shared secret, which is known to both the TACACS+ client and server. TACACS+ server
administrators SHOULD configure a shared secret with a minimum length of 16 characters. 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, and punctuation. Note that this security mechanism is best described as ‘obfuscation’ and not ‘encryption’ as it does not provide any meaningful integrity, privacy, or replay protection.

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
  "Indicates whether the Single Connection Mode is enabled for the server. By default, the Single Connection Mode is disabled.";
}

leaf timeout {
  type uint16 {
    range "1..max";
  }
}
5. Security Considerations

The YANG module specified in this document defines a schema for data that 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 Network Configuration Access Control Model (NACM) [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/tacacs-plus/server: This list contains the data nodes used to control the TACACS+ servers used by the device. Unauthorized access to this list could enable an attacker to assume complete control over the device by pointing to a compromised TACACS+ server, or to modify the counters to hide attacks against the device.

/system/tacacs-plus/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, it has been restricted using the "default-deny-all" access control defined in
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, and 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 10 of the TACACS+ protocol [RFC8907].

6. IANA Considerations

IANA has registered the following URI in the "ns" subregistry within the "IETF XML Registry" [RFC3688]:

Registrant Contact: The IESG.
XML: N/A, the requested URI is an XML namespace.

IANA has registered the following YANG module in the "YANG Module Names" registry [RFC7950]:

Name: ietf-system-tacacs-plus
Maintained by IANA: N
Prefix: sys-tcs-plus
Reference: RFC 9105

7. References

7.1. Normative References


7.2. Informative References


Appendix A. Example TACACS+ Authentication Configuration

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

```json
{
   "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"
         }
      ]
   }
}
```

Acknowledgments

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

Authors' Addresses

**Bo Wu (EDITOR)**
Huawei Technologies, Co., Ltd
Yuhua District
101 Software Avenue
Nanjing
Jiangsu, 210012
China
Email: lana.wubo@huawei.com
Guangying Zheng
Huawei Technologies, Co., Ltd
Yuhua District
101 Software Avenue
Nanjing
Jiangsu, 210012
China
Email: zhengguangying@huawei.com

Michael Wang (EDITOR)
Huawei Technologies, Co., Ltd
Yuhua District
101 Software Avenue
Nanjing
210012
China
Email: wangzitao@huawei.com