

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: October 23, 2013

A. Bierman
YumaWorks
M. Bjorklund
Tail-f Systems
April 21, 2013

YANG Data Model for System Management
draft-ietf-netmod-system-mgmt-06

Abstract

This document defines a YANG data model for the configuration and identification of the management system of a device.

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 October 23, 2013.

Copyright Notice

Copyright (c) 2013 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 (<http://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	3
1.1.	Terminology	3
1.1.1.	Terms	3
1.2.	Tree Diagrams	3
2.	Objectives	5
2.1.	System Identification	5
2.2.	System Time Management	5
2.3.	User Authentication	5
3.	System Data Model	6
3.1.	System Identification	6
3.2.	System Time Management	6
3.3.	DNS Resolver Model	6
3.4.	RADIUS Client Model	7
3.5.	User Authentication Model	7
3.5.1.	SSH Public Key Authentication	8
3.5.2.	Local User Password Authentication	8
3.5.3.	RADIUS Password Authentication	9
3.6.	System Control	9
4.	System YANG module	10
5.	IANA Considerations	26
6.	Security Considerations	27
7.	Change Log	29
7.1.	00-01	29
7.2.	01-02	29
7.3.	02-03	29
7.4.	03-04	29
7.5.	04-05	29
7.6.	05-06	30
8.	Normative References	31
	Authors' Addresses	33

1. Introduction

This document defines a YANG [[RFC6020](#)] data model for the configuration and identification of some common properties within the management system of a device.

Devices that are managed by NETCONF and perhaps other mechanisms have common properties that need to be configured and monitored in a standard way.

The "ietf-system" YANG module defined in this document provides the following features:

- o system administrative data configuration
- o system identification monitoring
- o system time-of-day configuration and monitoring
- o user authentication configuration
- o local users configuration

1.1. Terminology

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 [BCP 14](#), [[RFC2119](#)].

1.1.1. Terms

The following terms are used within this document:

- o system: This term refers to the embodiment of the entire set of management interfaces that a single NETCONF server is supporting

at a given moment. The set of physical entities managed by a single NETCONF server can be static or it can change dynamically.

[1.2.](#) Tree Diagrams

A simplified graphical representation of the data model is used in this document. The meaning of the symbols in these diagrams is as follows:

- o Brackets "[" and "]" enclose list keys.

- o Abbreviations before data node names: "rw" means configuration (read-write) and "ro" state data (read-only).
- o Symbols after data node names: "?" means an optional node and "*" denotes a "leaf-list".
- o Parentheses enclose choice and case nodes, and case nodes are also marked with a colon (":").
- o Ellipsis ("...") stands for contents of subtrees that are not shown.

[2.](#) Objectives

[2.1.](#) System Identification

There are many common properties used to identify devices, operating systems, software versions, etc. that need to be supported in the system data module. These objects are defined as operational data and the information returned by the server is intended to be specific to the device vendor.

Some user-configurable administrative strings are also provided, such as the system location and description.

[2.2.](#) System Time Management

The management of the date and time used by the system need to be supported. Use of one or more NTP servers to automatically set the system date and time need to be possible. Utilization of the Timezone database [[RFC6557](#)] also need to be supported. It should be possible for the server, as well as clients, to configure the system to use NTP.

[2.3.](#) User Authentication

The authentication mechanism need to support password authentication over RADIUS, to support deployment scenarios with centralized authentication servers. Additionally, local users need to be supported, for scenarios when no centralized authentication server exists, or for situations where the centralized authentication server cannot be reached from the device.

Since the mandatory transport protocol for NETCONF is SSH [[RFC6242](#)] the authentication model need to support SSH's "publickey" and "password" authentication methods [[RFC4252](#)].

The model for authentication configuration should be flexible enough to support authentication methods defined by other standard documents or by vendors. It should be possible for the server, as well as clients, to configure the system authentication properties.

[3.](#) System Data Model

[3.1.](#) System Identification

The data model for system identification has the following structure:

```
+--rw system
  +--rw contact?          string
  +--rw hostname?         inet:domain-name
  +--rw location?         string
  +--ro platform
    +--ro os-name?        string
    +--ro os-release?     string
    +--ro os-version?     string
    +--ro machine?        string
```

[3.2.](#) System Time Management

The data model for system time management has the following structure:

```
+--rw system
  +--rw clock
    | +--ro current-datetime?      yang:date-and-time
    | +--ro boot-datetime?        yang:date-and-time
    | +--rw (timezone)?
    |   +--:(timezone-location)
    |     | +--rw timezone-location?  string
    |     +--:(timezone-utc-offset)
    |       +--rw timezone-utc-offset? int16
  +--rw ntp
    +--rw enabled?      boolean
    +--rw server [address]
      +--rw association-type? enumeration
      +--rw address      inet:host
      +--rw enabled?     boolean
      +--rw iburst?      boolean
      +--rw prefer?      boolean
```

[3.3.](#) DNS Resolver Model

The data model for configuration of the DNS resolver has the following structure:

```
+--rw system
  +--rw dns
    +--rw search*    inet:domain-name
    +--rw server*    inet:ip-address
    +--rw options
      +--rw timeout?  uint8
      +--rw attempts? uint8
```

[3.4.](#) RADIUS Client Model

The data model for configuration of the RADIUS client has the following structure:

```

+--rw system
  +--rw radius
    +--rw server [address]
      |   +--rw address      inet:host
      |   +--rw (transport)
      |   |   +--:(udp)
      |   |   +--rw udp
      |   |   +--rw authentication-port?  inet:port-number
      |   |   +--rw shared-secret        string
      |   +--rw authentication-type?  identityref
    +--rw options
      +--rw timeout?    uint8
      +--rw attempts?  uint8

```

[3.5.](#) User Authentication Model

This document defines three authentication methods for use with NETCONF:

- o publickey for local users over SSH
- o password for local users over any transport
- o password for RADIUS users over any transport

Additional methods can be defined by other standard documents or by vendors.

This document defines two optional YANG features, "local-users" and "radius-authentication", which the server advertises to indicate support for configuring local users on the device, and support for using RADIUS for authentication, respectively.

The authentication parameters defined in this document are primarily used to configure authentication of NETCONF users, but MAY also be

based User Interface.

The data model for user authentication has the following structure:

```
+--rw system
  +--rw authentication
    +--rw user-authentication-order*  identityref
    +--rw user [name]
      +--rw name          string
      +--rw password?     crypt-hash
      +--rw ssh-key [name]
        +--rw name        string
        +--rw algorithm?  string
        +--rw key-data?   binary
```

[3.5.1.](#) SSH Public Key Authentication

If the NETCONF server advertises the "local-users" feature, configuration of local users and their SSH public keys is supported in the /system/authentication/user list.

Public key authentication is requested by the SSH client. If the "local-users" feature is supported, then when a NETCONF client starts an SSH session towards the server using the "publickey" authentication "method name" [[RFC4252](#)], the SSH server looks up the user name given in the SSH authentication request in the /system/authentication/user list, and verifies the key as described in [[RFC4253](#)].

[3.5.2.](#) Local User Password Authentication

If the NETCONF server advertises the "local-users" feature, configuration of local users and their passwords is supported in the /system/authentication/user list.

For NETCONF transport protocols that support password authentication, the leaf-list "user-authentication-order" is used to control if local user password authentication should be used.

In SSH, password authentication is requested by the client. Other NETCONF transport protocols MAY also support password authentication.

When local user password authentication is requested, the NETCONF transport looks up the user name provided by the client in the /system/ authentication/user list, and verifies the password.

[3.5.3.](#) RADIUS Password Authentication

If the NETCONF server advertises the "radius-authentication" feature, the device supports user authentication using RADIUS.

For NETCONF transport protocols that support password authentication, the leaf-list "user-authentication-order" is used to control if RADIUS password authentication should be used.

In SSH, password authentication is requested by the client. Other NETCONF transport protocols MAY also support password authentication.

[3.6.](#) System Control

Two protocol operations are included to restart or shutdown the system. The 'system-restart' operation can be used to restart the entire system (not just the NETCONF server). The 'system-shutdown' operation can be used to power off the entire system.

4. System YANG module

This YANG module imports YANG extensions from [\[RFC6536\]](#), and imports YANG types from [\[I-D.ietf-netmod-rfc6021-bis\]](#) and [\[I-D.ietf-netmod-iana-timezones\]](#). It also references [\[RFC1321\]](#), [\[RFC2865\]](#), [\[RFC3418\]](#), [\[RFC5607\]](#), [\[IEEE-1003.1-2008\]](#), and [\[FIPS.180-3.2008\]](#).

RFC Ed.: update the date below with the date of RFC publication and remove this note.

<CODE BEGINS> file "ietf-system@2013-04-21.yang"

```
module ietf-system {
  namespace "urn:ietf:params:xml:ns:yang:ietf-system";
  prefix "sys";

  import ietf-yang-types {
    prefix yang;
  }

  import ietf-inet-types {
    prefix inet;
  }

  import ietf-netconf-acm {
    prefix nacm;
  }

  import iana-timezones {
    prefix ianatz;
  }

  organization
    "IETF NETMOD (NETCONF Data Modeling Language) Working Group";

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

WG Chair: David Kessens
<mailto:david.kessens@nsn.com>

WG Chair: Juergen Schoenwaelder
<mailto:j.schoenwaelder@jacobs-university.de>

Editor: Andy Bierman
<mailto:andy@yumaworks.com>

Bierman & Bjorklund

Expires October 23, 2013

[Page 10]

Internet-Draft

YANG System Management

April 2013

Editor: Martin Bjorklund
<mailto:mbj@tail-f.com>;

description

"This module contains a collection of YANG definitions for the configuration and identification of the management system of a device.

Copyright (c) 2013 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.: replace XXXX with actual RFC number and remove this
// note.

// RFC Ed.: remove this note
// Note: extracted from [draft-ietf-netmod-system-mgmt-06.txt](#)

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

revision "2013-04-21" {
 description
 "Initial revision.";

```

reference
  "RFC XXXX: A YANG Data Model for System Management";
}

/*
 * Typedefs
 */

typedef crypt-hash {
  type string {
    pattern
      '$0$.*'
    + '|$1$[a-zA-Z0-9./]{1,8}$[a-zA-Z0-9./]{22}'
    + '|$5$(rounds=\d+)$?[a-zA-Z0-9./]{1,16}$[a-zA-Z0-9./]{43}'
    + '|$6$(rounds=\d+)$?[a-zA-Z0-9./]{1,16}$[a-zA-Z0-9./]{86}';
  }
}

```

description

"The crypt-hash type is used to store passwords using a hash function. The algorithms for applying the hash function and encoding the result are implemented in various UNIX systems as the function crypt(3).

A value of this type matches one of the forms:

```

$0$<clear text password>
$id>$<salt>$<password hash>
$id>$<parameter>$<salt>$<password hash>

```

The '\$0\$' prefix signals that the value is clear text. When such a value is received by the server, a hash value is calculated, and the string '\$<id>\$<salt>\$' or '\$<id>\$<parameter>\$<salt>\$' is prepended to the result. This value is stored in the configuration data store.

If a value starting with '\$<id>\$', where <id> is not '0', is received, the server knows that the value already represents a hashed value, and stores it as is in the data store.

When a server needs to verify a password given by a user, it finds the stored password hash string for that user, extracts the salt, and calculates the hash with the salt and given

password as input. If the calculated hash value is the same as the stored value, the password given by the client is correct.

This type defines the following hash functions:

id	hash function	feature
1	MD5	crypt-hash-md5
5	SHA-256	crypt-hash-sha-256
6	SHA-512	crypt-hash-sha-512

The server indicates support for the different hash functions by advertising the corresponding feature.";

```
reference
  "IEEE Std 1003.1-2008 - crypt() function
  Wikipedia: http://en.wikipedia.org/wiki/Crypt\_\(C\)
  RFC 1321: The MD5 Message-Digest Algorithm
  FIPS.180-3.2008: Secure Hash Standard";
}

/*
 * Features
```

```
*/

feature radius {
  description
    "Indicates that the device can be configured as a RADIUS
    client.";
  reference
    "RFC 2865: Remote Authentication Dial In User Service "
    + "(RADIUS)";
}

feature authentication {
  description
    "Indicates that the device can be configured
    to do authentication of users.";
}

feature local-users {
```

```

    if-feature authentication;
    description
        "Indicates that the device supports
        local user authentication.";
}

feature radius-authentication {
    if-feature radius;
    if-feature authentication;
    description
        "Indicates that the device supports user authentication over
        RADIUS.";
    reference
        "RFC 2865: Remote Authentication Dial In User Service (RADIUS)
        RFC 5607: Remote Authentication Dial-In User Service (RADIUS)
        Authorization for Network Access Server (NAS)
        Management";
}

feature crypt-hash-md5 {
    description
        "Indicates that the device supports the MD5
        hash function in 'crypt-hash' values";
    reference "RFC 1321: The MD5 Message-Digest Algorithm";
}

feature crypt-hash-sha-256 {
    description
        "Indicates that the device supports the SHA-256
        hash function in 'crypt-hash' values";
}

```

```

    reference "FIPS.180-3.2008: Secure Hash Standard";
}

feature crypt-hash-sha-512 {
    description
        "Indicates that the device supports the SHA-512
        hash function in 'crypt-hash' values";
    reference "FIPS.180-3.2008: Secure Hash Standard";
}

feature ntp {

```

```

    description
        "Indicates that the device can be configured
        to use one or more NTP servers to set the
        system date and time.";
}

feature timezone-location {
    description
        "Indicates that the local timezone on the device
        can be configured to use the TZ database
        to set the timezone and manage daylight savings time.";
    reference
        "TZ Database http://www.twinsun.com/tz/tz-link.htm
        Maintaining the Timezone Database
        RFC 6557 (BCP 175)";
}

/*
 * Identities
 */

identity authentication-method {
    description
        "Base identity for user authentication methods.";
}

identity radius {
    base authentication-method;
    description
        "Indicates user authentication using RADIUS.";
    reference
        "RFC 2865: Remote Authentication Dial In User Service (RADIUS)
        RFC 5607: Remote Authentication Dial-In User Service (RADIUS)
        Authorization for Network Access Server (NAS)
        Management";
}

```

```

identity local-users {
    base authentication-method;
    description
        "Indicates password-based authentication of locally

```



```

        configured users.";
    }

    identity radius-authentication-type {
        description
            "Base identity for RADIUS authentication types.";
    }

    identity radius-pap {
        base radius-authentication-type;
        description
            "The device requests PAP authentication from the RADIUS
            server.";
        reference
            "RFC 2865: Remote Authentication Dial In User Service";
    }

    identity radius-chap {
        base radius-authentication-type;
        description
            "The device requests CHAP authentication from the RADIUS
            server.";
        reference
            "RFC 2865: Remote Authentication Dial In User Service";
    }

    /*
     * Top-level container
     */

    container system {
        description
            "System group configuration.";

        leaf contact {
            type string {
                length "0..255";
            }
            description
                "The administrator contact information for the system.";
            reference
                "RFC 3418 - Management Information Base (MIB) for the
                Simple Network Management Protocol (SNMP)
                SNMPv2-MIB.sysContact";
        }
    }

```

```
}

leaf hostname {
  type inet:domain-name;
  description
    "The name of the host. This name can be a single domain
    label, or the fully qualified domain name of the host.";
}

leaf location {
  type string {
    length "0..255";
  }
  description
    "The system location.";
  reference
    "RFC 3418 - Management Information Base (MIB) for the
    Simple Network Management Protocol (SNMP)
    SNMPv2-MIB.sysLocation";
}

container platform {
  config false;
  description
    "Contains vendor-specific information for
    identifying the system platform and operating system.";
  reference
    "IEEE Std 1003.1-2008 - sys/utsname.h";

  leaf os-name {
    type string;
    description
      "The name of the operating system in use,
      for example 'Linux'";
    reference
      "IEEE Std 1003.1-2008 - utsname.sysname";
  }

  leaf os-release {
    type string;
    description
      "The current release level of the operating
      system in use. This string MAY indicate
      the OS source code revision.";
    reference
      "IEEE Std 1003.1-2008 - utsname.release";
  }
}
```

```
leaf os-version {
  type string;
  description
    "The current version level of the operating
    system in use. This string MAY indicate
    the specific OS build date and target variant
    information.";
  reference
    "IEEE Std 1003.1-2008 - utsname.version";
}

leaf machine {
  type string;
  description
    "A vendor-specific identifier string representing
    the hardware in use.";
  reference
    "IEEE Std 1003.1-2008 - utsname.machine";
}

}

container clock {
  description
    "Configuration and monitoring of the system
    date and time properties.";

  leaf current-datetime {
    type yang:date-and-time;
    config false;
    description
      "The current system date and time.";
  }

  leaf boot-datetime {
    type yang:date-and-time;
    config false;
    description
      "The system date and time when the NETCONF
      server last restarted.";
```

```

}

choice timezone {
  description
    "The system timezone information.";

  leaf timezone-location {
    if-feature timezone-location;

```

```

    type ianatz:iana-timezone;
    description
      "The TZ database location identifier string
      to use for the system, such as 'Europe/Stockholm'.";
  }

  leaf timezone-utc-offset {
    type int16 {
      range "-1500 .. 1500";
    }
    units "minutes";
    description
      "The number of minutes to add to UTC time to
      identify the timezone for this system.
      For example, 'UTC - 8:00 hours' would be
      represented as '-480'. Note that automatic
      daylight savings time adjustment is not provided,
      if this object is used.";
  }
}

container ntp {
  if-feature ntp;

  description
    "Configuration of the NTP client.";

  leaf enabled {
    type boolean;
    default true;
    description
      "Indicates that the system should attempt

```

```

        to synchronize the system clock with an
        NTP server from the 'ntp/server' list.";
    }

    list server {
        key address;
        description
            "List of NTP servers to use for
            system clock synchronization.  If '/system/ntp/enabled'
            is 'true', then the system will attempt to
            contact and utilize the specified NTP servers.";

        leaf association-type {
            type enumeration {
                enum server {

```

```

        description
            "Use client association mode.  This device
            will not provide synchronization to the
            configured NTP server.";
    }
    enum peer {
        description
            "Use symmetric active association mode.
            This device may provide synchronization
            to the configured NTP server.";
    }
    enum pool {
        description
            "Use client association mode with one or
            more of the NTP servers found by DNS
            resolution of the domain name given by
            the 'address' leaf.  This device will not
            provide synchronization to the servers.";
    }
    }
    default server;
    description
        "The desired association type for this NTP server.";
    }
    leaf address {
        type inet:host;

```

```

        description
            "The IP address or domain name of the NTP server.";
    }
    leaf enabled {
        type boolean;
        default true;
        description
            "Indicates whether this server is enabled for use or
            not.";
    }
    leaf iburst {
        type boolean;
        default false;
        description
            "Indicates whether this server should enable burst
            synchronization or not.";
    }
    leaf prefer {
        type boolean;
        default false;
        description
            "Indicates whether this server should be preferred

```

```

        or not.";
    }
}
}

container dns {
    description
        "Configuration of the DNS resolver.";

    leaf-list search {
        type inet:domain-name;
        ordered-by user;
        description
            "An ordered list of domains to search when resolving
            a host name.";
    }
    leaf-list server {
        type inet:ip-address;
        ordered-by user;

```

```

    description
      "Addresses of the name servers that the resolver should
       query.

       Implementations MAY limit the number of entries in this
       leaf list.";
  }
  container options {
    description
      "Resolver options. The set of available options has been
       limited to those that are generally available across
       different resolver implementations, and generally
       useful.";
    leaf timeout {
      type uint8 {
        range "1..max";
      }
      units "seconds";
      default "5";
      description
        "The amount of time the resolver will wait for a
         response from a remote name server before
         retrying the query via a different name server.";
    }
    leaf attempts {
      type uint8 {
        range "1..max";
      }
      default "2";
    }
  }

```

```

    description
      "The number of times the resolver will send a query to
       its name servers before giving up and returning an
       error to the calling application.";
  }
}

container radius {
  if-feature radius;

  description

```

```

    "Configuration of the RADIUS client.";

list server {
    key address;
    ordered-by user;
    description
        "List of RADIUS servers used by the device.";

    leaf address {
        type inet:host;
        description
            "The address of the RADIUS server.";
    }
    choice transport {
        mandatory true;
        description
            "The transport protocol specific parameters
            for this server. It is expected that new
            case statements will be added over time to
            support other transport protocols.";
        case udp {
            container udp {
                description
                    "Contains UDP specific configuration parameters
                    for RADIUS.";
                leaf authentication-port {
                    type inet:port-number;
                    default "1812";
                    description
                        "The port number of the RADIUS server.";
                }
                leaf shared-secret {
                    type string;
                    mandatory true;
                    nacm:default-deny-all;
                    description

```

```

        "The shared secret which is known to both the
        RADIUS client and server.";
reference
    "RFC 2865: Remote Authentication Dial In User
    Service";

```



```

    }
  }
}
leaf authentication-type {
  type identityref {
    base radius-authentication-type;
  }
  default radius-pap;
  description
    "The authentication type requested from the RADIUS
    server.";
}
}
container options {
  description
    "RADIUS client options.";

  leaf timeout {
    type uint8 {
      range "1..max";
    }
    units "seconds";
    default "5";
    description
      "The number of seconds the device will wait for a
      response from a RADIUS server before trying with a
      different server.";
  }
  leaf attempts {
    type uint8 {
      range "1..max";
    }
    default "2";
    description
      "The number of times the device will send a query to
      the RADIUS servers before giving up.";
  }
}
}

container authentication {
  nacm:default-deny-write;
}

```

```

if-feature authentication;

description
    "The authentication configuration subtree.";

leaf-list user-authentication-order {
    type identityref {
        base authentication-method;
    }
    must '(. != "sys:radius" or ../../radius/server)' {
        error-message
            "When 'radius' is used, a RADIUS server"
            + " must be configured.";
        description
            "When 'radius' is used as an authentication method,
            a RADIUS server must be configured.";
    }
    ordered-by user;

    description
        "When the device authenticates a user with
        a password, it tries the authentication methods in this
        leaf-list in order.  If authentication with one method
        fails, the next method is used.  If no method succeeds,
        the user is denied access.

        If the 'radius-authentication' feature is advertised by
        the NETCONF server, the 'radius' identity can be added to
        this list.

        If the 'local-users' feature is advertised by the
        NETCONF server, the 'local-users' identity can be
        added to this list.";
}

list user {
    if-feature local-users;
    key name;
    description
        "The list of local users configured on this device.";

    leaf name {
        type string;
        description
            "The user name string identifying this entry.";
    }
    leaf password {
        type crypt-hash;

```

Internet-Draft

YANG System Management

April 2013

```
        description
            "The password for this entry.";
    }
    list ssh-key {
        key name;
        description
            "A list of public SSH keys for this user.";
        reference
            "RFC 4253: The Secure Shell (SSH) Transport Layer
            Protocol";

        leaf name {
            type string;
            description
                "An arbitrary name for the ssh key.";
        }
        leaf algorithm {
            type string;
            description
                "The public key algorithm name for this ssh key.

                Valid values are the values in the IANA Secure Shell
                (SSH) Protocol Parameters registry, Public Key
                Algorithm Names";
            reference
                "IANA Secure Shell (SSH) Protocol Parameters registry,
                Public Key Algorithm Names";
        }
        leaf key-data {
            type binary;
            description
                "The binary key data for this ssh key.";
        }
    }
}

rpc set-current-datetime {
    nacm:default-deny-all;
    description
        "Set the /system/clock/current-datetime leaf"
```

to the specified value.

If the system is using NTP (i.e., /system/ntp/enabled is set to 'true'), then this operation will fail with error-tag 'operation-failed', and error-app-tag value of 'ntp-active'";

```
input {
  leaf current-datetime {
    type yang:date-and-time;
    mandatory true;
    description
      "The current system date and time.";
  }
}

rpc system-restart {
  nacm:default-deny-all;
  description
    "Request that the entire system be restarted immediately.
    A server SHOULD send an rpc reply to the client before
    restarting the system.";
}

rpc system-shutdown {
  nacm:default-deny-all;
  description
    "Request that the entire system be shut down immediately.
    A server SHOULD send an rpc reply to the client before
    shutting down the system.";
}

}

<CODE ENDS>
```

5. IANA Considerations

This document registers one URI in the IETF XML registry [[RFC3688](#)]. Following the format in [RFC 3688](#), the following registration is requested to be made.

URI: urn:ietf:params:xml:ns:yang:ietf-system
Registrant Contact: The NETMOD WG of the IETF.
XML: N/A, the requested URI is an XML namespace.

This document registers one YANG module in the YANG Module Names registry [[RFC6020](#)].

name: ietf-system
namespace: urn:ietf:params:xml:ns:yang:ietf-system
prefix: sys
reference: RFC XXXX

6. Security Considerations

The YANG module defined in this memo is designed to be accessed via the NETCONF protocol [[RFC6241](#)]. The lowest NETCONF layer is the secure transport layer and the mandatory-to-implement secure transport is SSH [[RFC6242](#)].

There are a number of data nodes defined in this YANG module which 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:

- o /system/clock/timezone: This choice contains the objects used to control the timezone used by the device.
- o /system/ntp: This container contains the objects used to control the Network Time Protocol servers used by the device.

- o /system/dns: This container contains the objects used to control the Domain Name System servers used by the device.
- o /system/radius: This container contains the objects used to control the Remote Authentication Dial-In User Service servers used by the device.
- o /system/authentication/user-authentication-order: This leaf controls how user login attempts are authenticated by the device.
- o /system/authentication/user: This list contains the local users enabled on the system.

Some of the readable data nodes in this YANG module may be considered sensitive or vulnerable in some network environments. It is thus important to control read access (e.g., via get, get-config, or notification) to these data nodes. These are the subtrees and data nodes and their sensitivity/vulnerability:

- o /system/platform: This container has objects which may help identify the specific NETCONF server and/or operating system implementation used on the device.

Some of the RPC operations in this YANG module may be considered sensitive or vulnerable in some network environments. It is thus important to control access to these operations. These are the operations and their sensitivity/vulnerability:

- o set-current-datetime: Changes the current date and time on the device.
- o system-restart: Reboots the device.
- o system-shutdown: Shuts down the device.

[7.](#) Change Log

-- RFC Ed.: remove this section before publication.

[7.1.](#) 00-01

- o added configuration-source identities

- o added configuration-source leaf to ntp and dns (via grouping) to choose configuration source
- o added association-type, iburst, prefer, and true leafs to the ntp-server list
- o extended the ssh keys for a user to a list of keys. support all defined key algorithms, not just dsa and rsa
- o clarified timezone-utc-offset description-stmt
- o removed '/system/ntp/server/true' leaf from data model

[7.2.](#) 01-02

- o added default-stmts to ntp-server/iburst and ntp-server/prefer leafs
- o changed timezone-location leaf to use iana-timezone typedef instead of a string

[7.3.](#) 02-03

- o removed configuration-source identities and leafs

[7.4.](#) 03-04

- o removed ndots dns resolver option
- o added radius-authentication-type identity, and identities for pap and chap, and a leaf to control which authentication type to use when communicating with the radius server
- o made 0 an invalid value for timeouts and attempts

[7.5.](#) 04-05

- o updated tree diagram explanation text

[7.6.](#) 05-06

- o changed ntp/use-ntp to ntp/enabled
- o changed ntp/ntp-server to ntp/server
- o removed /system/platform/nodename leaf
- o changed /system/name to /system/hostname
- o simplified must expression in user-authentication-order
- o added optional rounds to sha hash definition
- o clarified the crypt-hash description
- o clarified ntp descriptions
- o clarified YANG module description to indicate that some system properties are supported, not the entire system
- o clarified that system identification values are vendor specific, not the data node objects
- o clarified sec. 2.2 and 2.3 to indicate that the server should also be capable of configuring these properties
- o changed /system/dns/search from inet:host to inet:domain-name
- o changed [RFC6021](#) reference to 6021-bis
- o changed /system/platform/nodename to /system/platform/hostname
- o changed /system/radius/server/{leafs} to be within a choice and 'udp' case statement so other transport specific parameters can augment this list or they can be added by the WG to a future version of this module. {leafs} are authentication-port and shared-secret.
- o updated YANG tree diagrams for objects added in -05 and -06

Internet-Draft

YANG System Management

April 2013

8. Normative References

[FIPS.180-3.2008]

National Institute of Standards and Technology, "Secure Hash Standard", FIPS PUB 180-3, October 2008, <http://csrc.nist.gov/publications/fips/fips180-3/fips180-3_final.pdf>.

[I-D.ietf-netmod-iana-timezones]

Lange, J., "IANA Timezone Database YANG Module", [draft-ietf-netmod-iana-timezones-00](#) (work in progress), July 2012.

[I-D.ietf-netmod-rfc6021-bis]

Schoenwaelder, J., "Common YANG Data Types", [draft-ietf-netmod-rfc6021-bis-01](#) (work in progress), March 2013.

[IEEE-1003.1-2008]

Institute of Electrical and Electronics Engineers, "POSIX.1-2008", IEEE Standard 1003.1, March 2008.

[RFC1321] Rivest, R., "The MD5 Message-Digest Algorithm", [RFC 1321](#), April 1992.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.

[RFC2865] Rigney, C., Willens, S., Rubens, A., and W. Simpson, "Remote Authentication Dial In User Service (RADIUS)", [RFC 2865](#), June 2000.

[RFC3418] Presuhn, R., "Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)", STD 62, [RFC 3418](#), December 2002.

[RFC3688] Mealling, M., "The IETF XML Registry", [BCP 81](#), [RFC 3688](#), January 2004.

[RFC4252] Ylonen, T. and C. Lonvick, "The Secure Shell (SSH) Authentication Protocol", [RFC 4252](#), January 2006.

[RFC4253] Ylonen, T. and C. Lonvick, "The Secure Shell (SSH)

Transport Layer Protocol", [RFC 4253](#), January 2006.

- [RFC5607] Nelson, D. and G. Weber, "Remote Authentication Dial-In User Service (RADIUS) Authorization for Network Access Server (NAS) Management", [RFC 5607](#), July 2009.

Bierman & Bjorklund

Expires October 23, 2013

[Page 31]

Internet-Draft

YANG System Management

April 2013

- [RFC6020] Bjorklund, M., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", [RFC 6020](#), October 2010.
- [RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", [RFC 6241](#), June 2011.
- [RFC6242] Wasserman, M., "Using the NETCONF Protocol over Secure Shell (SSH)", [RFC 6242](#), June 2011.
- [RFC6536] Bierman, A. and M. Bjorklund, "Network Configuration Protocol (NETCONF) Access Control Model", [RFC 6536](#), March 2012.
- [RFC6557] Lear, E. and P. Eggert, "Procedures for Maintaining the Time Zone Database", [BCP 175](#), [RFC 6557](#), February 2012.

Authors' Addresses

Andy Bierman
YumaWorks

Email: andy@yumaworks.com

Martin Bjorklund
Tail-f Systems

Email: mbj@tail-f.com

