Network Inventory YANG
Internet-Draft
Intended status: Standards Track
Expires: 23 April 2026

B. Wu
Huawei
C. Zhou
China Mobile
Q. Wu
Huawei
M. Boucadair
Orange

20 October 2025

A YANG Network Data Model of Network Inventory Software Extensions
draft-ietf-ivy-network-inventory-software-02

Abstract

This document extends the base Network Inventory YANG model to support non-physical network elements (NEs), such as controllers, virtual routers, and virtual firewalls, as well as software components like platform operating systems and software modules. In addition to the software revisions and patches already defined in the base model, this extension introduces software status and time stamp information.

Discussion Venues

This note is to be removed before publishing as an RFC.

Discussion of this document takes place on the Network Inventory YANG mailing list (inventory-yang@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/inventory-yang/.

Source for this draft and an issue tracker can be found at https://github.com/ietf-ivy-wg/network-inventory-software.

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/.

Copyright Notice

Table of Contents

1. Introduction

The Network Inventory consists of the physical and non-physical network elements (NEs), hardware components, firmware components, and software components on a managed network domain. The non-physical network elements (NEs) are network devices that support network protocols and functions, e.g., routers, firewalls, and controllers, which can reside in any network or compute devices, such as servers in Data Center (DC), server-based virtual machines (VMs), or server-based containers.

[I-D.ietf-ivy-network-inventory-yang] defines the base Network Inventory YANG model for physical network element (NE) and hardware components of NEs. Examples of hardware components could be rack, shelf, slot, board and physical port.

The management of non-physical NE and software components information is similar to the management of physical NE and hardware information. For example, inventory data, including product names, serial numbers, etc. are also applicable. This document defines a network inventory software extension YANG model. In addition to inheriting the common inventory attributes of the base network inventory model, this document also adds some software-specific attributes of non-physical NEs (such as controllers, virtual routers, and virtual firewalls) and software components (such as operating system, software modules, BIOS, and boot loader).

The Network Inventory software extension model is classified as a network model (Section 4 of [RFC8309]).

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

1.1. Editorial Note (To be removed by RFC Editor)

Note to the RFC Editor: This section is to be removed prior to publication.

This document contains placeholder values that need to be replaced with finalized values at the time of publication. This note summarizes all of the substitutions that are needed.

Please apply the following replacements:

* XXXX --> the assigned RFC number for this I-D

* AAAA --> the assigned RFC number for [I-D.ietf-ivy-network-inventory-yang]

1.2. Terminology and Notations

The following terms are defined in [RFC7950] and are not redefined here:

* client
* server
* augment
* data model
* data node The following terms are defined in [RFC6241] and are not redefined here:
* configuration data
* state data The tree diagram used in this document follows the notation defined in [RFC8340]..

Also, this document uses terms defined in [I-D.ietf-ivy-network-inventory-yang].

2. Requirements Language

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.

3. Relationship to Other YANG Data Models

The base network inventory model supports the software versions of NEs and software versions of hardware components. This document adds more software component identifiers (e.g. platformos, software patch) and more NE types (e.g. software NE, virtual NE) to provide enhanced software information on the NE to facilitate software compatibility check.

Figure 1 depicts the relationship between the Software Extension model and the base network inventory model. The Software Extension network inventory model enhances the model defined in the base network inventory model with more software specific attributes.

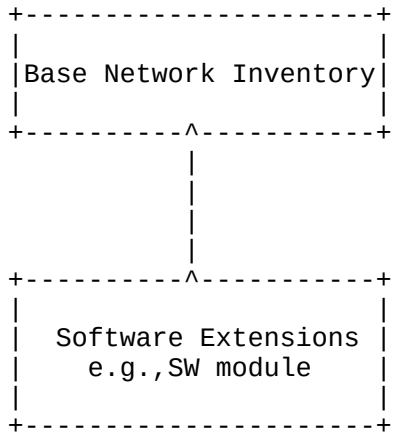

Figure 1: Relationship of SW Extension Model to the Base Inventory Model

4. Model Overview

The tree diagram in Figure 2 provides an overview of the data model for "ietf-network-inventory-sw-ext" module.

```
module: ietf-network-inventory-sw-ext
  augment /nwi:network-inventory/nwi:network-elements
            /nwi:network-element/nwi:software-rev:
    +--ro status?              identityref
    +--ro installation-time?   yang:date-and-time
    +--ro activation-time?     yang:date-and-time
  augment /nwi:network-inventory/nwi:network-elements
            /nwi:network-element/nwi:software-rev/nwi:patch:
    +--ro status?              identityref
    +--ro installation-time?   yang:date-and-time
    +--ro activation-time?     yang:date-and-time
  augment /nwi:network-inventory/nwi:network-elements
            /nwi:network-element/nwi:components/nwi:component
            /nwi:software-rev:
    +--ro status?              identityref
    +--ro installation-time?   yang:date-and-time
    +--ro activation-time?     yang:date-and-time
  augment /nwi:network-inventory/nwi:network-elements
            /nwi:network-element/nwi:components/nwi:component
            /nwi:software-rev/nwi:patch:
    +--ro status?              identityref
    +--ro installation-time?   yang:date-and-time
    +--ro activation-time?     yang:date-and-time
```

Figure 2: YANG Tree of Software Extensions

5. Non-physical Network Elements

In the base Network Inventory YANG model, "ne-type" is a YANG identity that describes the type of the network element and only the "physical-network-element" identity" is defined. This document adds non-physical NE identity, such as "ne-software", "ne-virtual", and "ne-container".

The base Network Inventory model also defines common inventory attributes, including the software version, patch versions, product name, and serial number. The data is also applicable to non-physical NEs.

The Network Inventory software extension mode defines some new software attributes, consisting of software status, installation time, and activation time.

6. Software components

Software components refer to the software installed on the NE, such as operating system, software modules, BIOS, and boot loaders.

Similar to the common inventory attributes of NEs, the common attributes of software components (such as software revisions, patch revisions, product name, and serial number) are also applicable to software components. For software revisions and patch revisions, the base inventory (Section 4 of [I-D.ietf-ivy-network-inventory-yang]) defines the "list" of "software-rev" and the "list" of "patch". For example, on a router, software components may include a routing protocol package (e.g., " foo-rt-protocol-suite"), or a firmware module for a line card (e.g., " foo-lc-fw-21.5.3").

If more detailed installation and activation information is needed— such as whether a component is active, pending-reboot, or rollback-eligible, along with its install time or activation time stamp, the extension attributes of software components can be used.

7. YANG Data model for Network Inventory Software Extensions

The "ietf-network-inventory-sw-ext" module uses types defined in [RFC6991], [I-D.ietf-ivy-network-inventory-yang].

```
<CODE BEGINS> file "ietf-network-inventory-sw-ext@2025-10-20.yang"
module ietf-network-inventory-sw-ext {
  yang-version 1.1;
  namespace
    "urn:ietf:params:xml:ns:yang:ietf-network-inventory-sw-ext";
  prefix nwis;

  import ietf-yang-types {
    prefix yang;
    reference
      "RFC 6991: Common YANG Data Types";
  }
  import ietf-network-inventory {
    prefix nwi;
    reference
      "RFCAAAA: A YANG Data Model for Network Inventory";
  }

  organization
    "IETF Network Inventory YANG (ivy) Working Group";
  contact
    "WG Web:   <https://datatracker.ietf.org/wg/ivy>
     WG List:  <mailto:inventory-yang@ietf.org>

     Editor: Bo Wu
             <lana.wubo@huawei.com>
     Editor: Cheng Zhou
          <zhouchengyjy@chinamobile.com>
     Editor: Qin Wu
             <bill.wu@huawei.com>
     Editor: Mohamed Boucadair
             <mohamed.boucadair@orange.com>";
  description
    "This YANG module defines a model for network inventory software
     extensions.

     Copyright (c) 2025 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 Revised 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 XXXX; see
     the RFC itself for full legal notices.
```

```
        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.";

      revision 2025-10-20 {
        description
          "Initial version";
        reference
          "RFC XXXX: A YANG Data Model for Network Inventory Software
                     Extensions.";
      }

      identity ne-nonphysical {
        base nwi:ne-type;
        description
          "Any network element implemented purely in software.
           It performs protocol or forwarding functions but
           does not correspond to a distinguishable hardware
           chassis. It can be hosted on a bare-metal server,
           VM, container, or cloud instance.";
      }

      identity ne-software {
        base ne-nonphysical;
        description
          "Software NE that runs directly on a host OS
           (a.k.a. bare-metal deployment) or hypervisor.
           Examples of software NEs are network controllers.";
      }

      identity ne-virtual {
        base ne-nonphysical;
        description
          "Virtual NE instantiated inside a virtual-machine
           (VM). Provides virtual network function (VNF) implementations
           such as vRouter, vFirewall, vPE.";
      }

      identity ne-container {
        base ne-nonphysical;
        description
          "Container NE packaged as CNF (Containerised Network
           Function). Runs under Docker/K8s.";
      }

      identity software-component {
```

```
  base nwi:non-hardware-component-class;
  description
    "Base identity for software components in a managed device.";
}

identity operating-system {
  base software-component;
  description
    "Operating system software type.";
}

identity bios {
  base software-component;
  description
    "BIOS or UEFI firmware image responsible for hardware
     initialisation and secure boot.";
}

identity boot-loader {
  base software-component;
  description
    "Software layer responsible for loading and booting the
     device OS or network OS.";
}

identity software-module {
  base software-component;
  description
    "Installable unit smaller than a full OS image,
     e.g. feature package.";
}

identity software-status {
  description
    "Base identity for software status.";
}

identity software-installed {
  base software-status;
  description
    "Software status is Installed.";
}

identity software-activated {
  base software-status;
  description
    "Software status is Activated.";
}
```

```
  grouping software-info-grouping {
    description
      "Specific attributes applicable to software.";
    leaf status {
      type identityref {
        base software-status;
      }
      description
        "Software status.";
    }
    leaf installation-time {
      type yang:date-and-time;
      description
        "Time when the software or patch revision was
         first installed.";
    }
    leaf activation-time {
      type yang:date-and-time;
      description
        "Time when the currently installed revision became active
         (i.e., was rebooted into).";
    }
  }

  /* Main blocks */

  augment "/nwi:network-inventory/nwi:network-elements"
        + "/nwi:network-element/nwi:software-rev" {
    description
      "Adds installation-/activation-time, status, etc. to the base
       NE software revision.";
    uses software-info-grouping;
  }

  augment "/nwi:network-inventory/nwi:network-elements"
        + "/nwi:network-element/nwi:software-rev/nwi:patch" {
    description
      "Adds installation-/activation-time, status, etc. to the patch
       level.";
    uses software-info-grouping;
  }

  augment "/nwi:network-inventory/nwi:network-elements/"
        + "nwi:network-element/nwi:components/nwi:component/"
        + "nwi:software-rev" {
    description
      "Extends components, such as line-card/CPU/etc.
       software revision with timestamp and state information.";
```

```
      uses software-info-grouping;
    }

    augment "/nwi:network-inventory/nwi:network-elements/"
          + "nwi:network-element/nwi:components/nwi:component/"
          + "nwi:software-rev/nwi:patch" {
      description
        "Applies the software-info attributes to component-level
         patches.";
      uses software-info-grouping;
    }
  }
  <CODE ENDS>
```

8. Security Considerations

This section uses the template described in Section 3.7 of [I-D.ietf-netmod-rfc8407bis].

The "ietf-network-inventory-sw-ext" YANG module defines a data model that is designed to be accessed via YANG-based management protocols, such as NETCONF NETCONF [RFC6241] or RESTCONF [RFC8040]. These YANG-based management protocols (1) have to use a secure transport layer (e.g., SSH [RFC4252], TLS [RFC8446], and QUIC [RFC9000]) and (2) have to use mutual authentication.

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.

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. Specifically, the following subtrees and data nodes have particular sensitivities/ vulnerabilities:

* "/nwi:network-elements/network-element/software-rev"

  This subtree reports the software information for all the network elements and their hardware components deployed within the network as well as of the software modules being active on these network elements and components. This may reveal software versions or unpatched vulnerabilities.

9. IANA Considerations

IANA is requested to register the following URI in the "ns" subregistry within the "IETF XML Registry" [RFC3688]:

URI: urn:ietf:params:xml:ns:yang:ietf-network-inventory-sw-ext
Registrant Contact: The IESG.
XML: N/A; the requested URI is an XML namespace.

IANA is requested to register the following YANG module in the "YANG Module Names" registry [RFC6020] within the "YANG Parameters" registry group:

Name: ietf-network-inventory-sw-ext
Namespace: urn:ietf:params:xml:ns:yang:ietf-network-inventory-sw-ext
Prefix: nwis
Maintained by IANA? N
Reference: RFC XXXX

10. References

10.1. Normative References

[I-D.ietf-ivy-network-inventory-yang]
Yu, C., Belotti, S., Bouquier, J., Peruzzini, F., and P. Bedard, "A Base YANG Data Model for Network Inventory", Work in Progress, Internet-Draft, draft-ietf-ivy-network-inventory-yang-11, 14 October 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-ivy-network-inventory-yang-11>.

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

[RFC6020] Bjorklund, M., Ed., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, DOI 10.17487/RFC6020, October 2010, <https://www.rfc-editor.org/rfc/rfc6020>.

[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/rfc/rfc6241>.

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

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

[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/rfc/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/rfc/rfc8342>.

10.2. Informative References

[I-D.ietf-netmod-rfc8407bis] Bierman, A., Boucadair, M., and Q. Wu, "Guidelines for Authors and Reviewers of Documents Containing YANG Data Models", Work in Progress, Internet-Draft, draft-ietf-netmod-rfc8407bis-28, 5 June 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-netmod-rfc8407bis-28>.

[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/rfc/rfc2119>.

[RFC4252] Ylonen, T. and C. Lonvick, Ed., "The Secure Shell (SSH) Authentication Protocol", RFC 4252, DOI 10.17487/RFC4252, January 2006, <https://www.rfc-editor.org/rfc/rfc4252>.

[RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017, <https://www.rfc-editor.org/rfc/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/rfc/rfc8174>.

[RFC8309] Wu, Q., Liu, W., and A. Farrel, "Service Models Explained", RFC 8309, DOI 10.17487/RFC8309, January 2018, <https://www.rfc-editor.org/rfc/rfc8309>.

[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/rfc/rfc8340>.

[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/rfc/rfc8446>.

[RFC9000] Iyengar, J., Ed. and M. Thomson, Ed., "QUIC: A UDP-Based Multiplexed and Secure Transport", RFC 9000, DOI 10.17487/RFC9000, May 2021, <https://www.rfc-editor.org/rfc/rfc9000>.

Appendix A. Examples of Software Attributes

This appendix provides some examples of software attributes implementations and how they can be modeled using the “ietf-network-inventory-sw-ext” module defined.

This appendix illustrates, by means of two typical scenarios, how to populate the software-specific nodes defined in ietf-network-inventory-sw-ext and explains the common values that can be used.

Scenario 1: Whole-device base software (example-os) plus hot patches (P3 already activated, P4 installed but not yet activated).

Scenario 2: Line-card programmable forwarding image (example-fpga-image) plus its patch (2.1.0.P1 installed and awaiting activation).

=============== NOTE: ’’ line wrapping per RFC 8792 ================

{ "ietf-network-inventory:network-inventory": { "network-elements": { "network-element": [{ "ne-id": "example:NE-01", "software-rev": [{ "name": "example:ne-os", "revision": "7.9.2", "ietf-network-inventory-sw-ext:status": "software-\ activated", "ietf-network-inventory-sw-ext:installation-time": "\ 2024-08-01T12:00:00Z", "ietf-network-inventory-sw-ext:activation-time": "2024\ -08-01T12:05:00Z", "patch": [{ "revision": "P3", "ietf-network-inventory-sw-ext:status": "software-\ activated", "ietf-network-inventory-sw-ext:installation-time"\ : "2024-09-15T10:30:00Z", "ietf-network-inventory-sw-ext:activation-time": "\ 2024-09-15T10:32:00Z" }, { "revision": "P4", "ietf-network-inventory-sw-ext:status": "software-\ installed", "ietf-network-inventory-sw-ext:installation-time"\ : "2024-10-01T14:00:00Z", "ietf-network-inventory-sw-ext:activation-time": \ null }] }], "components": { "component": [{ "component-id": "example:lpu/1/0", "class": "iana-hardware:module", "software-rev": [{ "name": "example-fp-image", "revision": "2.1.0", "ietf-network-inventory-sw-ext:status": "\ software-activated", "ietf-network-inventory-sw-ext:installation-time\ ": "2024-08-01T12:00:00Z", "ietf-network-inventory-sw-ext:activation-time"\ : "2024-08-01T12:06:00Z", "patch": [{ "revision": "2.1.0.P1", "ietf-network-inventory-sw-ext:status": "\ software-installed",

"ietf-network-inventory-sw-ext:installation-\ time":
"2024-10-01T14:10:00Z", "activation-time": null }] }] }] } }] } }
}

Acknowledgments

The authors would like to thank Prasenjit Manna,Phil Bedard, Diego R. Lopez, Italo Busi, and many others for their helpful comments and suggestions.

Contributors

Yao Zhao
Huawei
Email: zhaoyao.zhaoyao@huawei.com

Authors' Addresses

Bo Wu
Huawei
Email: lana.wubo@huawei.com

Cheng Zhou
China Mobile
Email: zhouchengyjy@chinamobile.com

Qin Wu
Huawei
Email: bill.wu@huawei.com

Mohamed Boucadair
Orange
Email: mohamed.boucadair@orange.com