

SAVI
Internet-Draft
Intended status: Standards Track
Expires: June 26, 2016

W. Meng
C. Wang
ZTE Corporation
December 24, 2015

**YANG Data Model for SAVI
draft-meng-savi-yang-01**

Abstract

This document defines a YANG data model for SAVI.

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 June 26, 2016.

Copyright Notice

Copyright (c) 2015 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
2.	Convention and Terminology	4
3.	Design of the Data Model	5
4.	SAVI YANG Model	6
5.	Security Considerations	11
6.	Acknowledgements	12
7.	Normative References	13
	Authors' Addresses	14

1. Introduction

IETF netmod WG has developed a general data model for NETCONF protocol, YANG data model [[RFC6020](#)].

This document defines a YANG data model for the configuration and management of SAVI. With this model, the operators can configure and manage the devices by using NETCONF.

2. Convention and Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [[RFC2119](#)].

3. Design of the Data Model

```
module: ietf-savi
  +--rw savi
    +--rw global
      | +--rw ip-version          uint8
      | +--rw enable              boolean
      | +--rw arrange-mode?      uint8
      | +--rw max-dad-delay?     uint16
      | +--rw max-dad-prepare-delay? uint16
      | +--rw max-ns-delay?     uint16
      | +--rw dhcp-only          boolean
    +--rw interface
      | +--rw interface* [index]
      |   +--rw index            uint32
      |   +--rw interface-name   string
      |   +--rw enable           boolean
      |   +--rw validation-status uint8
      |   +--rw filtering-number  uint8
    +--rw binding
      | +--rw binding-info* [ip-addr]
      |   +--rw ip-addr          inet:ipv6-address
      |   +--rw interface-index  uint32
      |   +--rw ip-addr-type     uint8
      |   +--rw type              uint8
      |   +--rw mac-addr          yang:mac-address
      |   +--rw status            uint8
      |   +--rw lifetime          uint32
      |   +--rw row-status        uint8
    +--rw filtering
      +--rw filtering-info* [interface-index]
        +--rw ip-addr?          inet:ipv6-address
        +--rw interface-index   uint32
        +--rw ip-addr-type      uint8
        +--rw mac-addr          yang:mac-address
```


4. SAVI YANG Model

This module imports typedefs from [[RFC6991](#)].

```
<CODE BEGINS> file "ietf-savi@2015-12-24.yang"

module ietf-savi {
  namespace "urn:ietf:params:xml:ns:yang:ietf-savi";
  prefix "savi";

  import ietf-interfaces {
    prefix "if";
  }

  import ietf-inet-types {
    prefix "inet";
  }

  import ietf-yang-types {
    prefix yang;
  }

  organization
    "IETF SAVI Working Group";
  contact
    "
      Editor:   Wei Meng
               <mailto:meng.wei2@zte.com.cn>
      Editor:   Cui Wang
               <mailto:wang.cui1@zte.com.cn>
    ";
  description "The YANG module defines a generic configuration model for
SAVI";

  revision 2015-12-24 {
    description "01 revision.";
    reference "draft-meng-savi-yang-01";
  }

  revision 2015-07-06 {
    description "Initial revision";
    reference "draft-meng-savi-yang-00";
  }

  /* Configuration Data */

  container savi {
```



```
description 'Configure SAVI parameters.' ;

container global{
  leaf ip-version {
    type uint8 ;
    mandatory true ;
    description 'IP Version.' ;
  }
  leaf enable {
    type boolean ;
    mandatory true ;
    description 'Enable.' ;
  }
  leaf arrange-mode {
    type uint8 ;
    description 'Arrange mode.' ;
  }
  leaf max-dad-delay {
    type uint16 {
      range 1..4094 {
        description 'The value of delay time,it must between 1 to 409'
          + '4.'
        ;
      }
    }
    description 'Max DAD delay time.' ;
  }
  leaf max-dad-prepare-delay {
    type uint16 {
      range 1..4094 {
        description 'The value of delay time,it must between 1 to 409'
          + '4.'
        ;
      }
    }
    description 'Max DAD prepare delay time.' ;
  }
  leaf max-ns-delay {
    type uint16 {
      range 1..4094 {
        description 'The value of delay time,it must between 1 to 409'
          + '4.'
        ;
      }
    }
    description 'Max NS delay time.' ;
  }
  leaf dhcp-only {
```



```
        type boolean ;
        mandatory true ;
        description 'DHCP ONLY.' ;
    }
    description 'Global configuration.' ;
}

container interface {
    list interface {
        key index ;
        leaf index {
            type uint32 {
                range 1..16777215 {
                    description 'The value of index,it must between 1 to
16777215.' ;
                }
            }
            description 'Interface index.' ;
        }
        leaf interface-name {
            type string;
            mandatory true;
            description 'Interface name.' ;
        }
        leaf enable {
            type boolean;
            mandatory true;
            description 'Enable.' ;
        }
        leaf validation-status {
            type uint8;
            mandatory true;
            description 'Validation status.' ;
        }
        leaf filtering-number {
            type uint8;
            mandatory true;
            description 'Filtering number.' ;
        }
        description 'Interface.' ;
    }
    description 'Interface.' ;
}

container binding {
    list binding-info {
        key ip-addr ;
        leaf ip-addr {
```

```
type inet:ipv6-address;
```

```
        description 'IP address.' ;
    }
    leaf interface-index {
        type uint32;
        mandatory true;
        description 'Interface index.' ;
    }
    leaf ip-addr-type {
        type uint8;
        mandatory true;
        description 'IP Address type.' ;
    }
    leaf type {
        type uint8;
        mandatory true;
        description 'Binding type.' ;
    }
    leaf mac-addr {
        type yang:mac-address;
        mandatory true;
        description 'Mac Address.' ;
    }
    leaf status {
        type uint8;
        mandatory true;
        description 'The status of binding information.' ;
    }
    leaf lifetime {
        type uint32;
        mandatory true;
        description 'The lifetime of binding information.' ;
    }
    leaf row-status {
        type uint8;
        mandatory true;
        description 'The row status of binding information.' ;
    }
    description 'Binding information.' ;
}
description 'Binding information.' ;
}

container filtering {
    list filtering-info {
        key interface-index;
        leaf ip-addr {
            type inet:ipv6-address;
            description 'IP address.' ;
        }
    }
}
```



```
    }
    leaf interface-index {
      type uint32;
      mandatory true;
      description 'Interface index.' ;
    }
    leaf ip-addr-type {
      type uint8;
      mandatory true;
      description 'IP Address type.' ;
    }
    leaf mac-addr {
      type yang:mac-address;
      mandatory true;
      description 'Mac address.' ;
    }
    description 'Filtering information.' ;
  }
  description 'Filtering information.' ;
}
}
```

<CODE ENDS>

5. Security Considerations

This document has no additional security considerations.

6. Acknowledgements

7. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), DOI 10.17487/[RFC2119](#), March 1997, <<http://www.rfc-editor.org/info/rfc2119>>.
- [RFC6020] Bjorklund, M., Ed., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", [RFC 6020](#), DOI 10.17487/RFC6020, October 2010, <<http://www.rfc-editor.org/info/rfc6020>>.
- [RFC6991] Schoenwaelder, J., Ed., "Common YANG Data Types", [RFC 6991](#), DOI 10.17487/RFC6991, July 2013, <<http://www.rfc-editor.org/info/rfc6991>>.

Authors' Addresses

Wei Meng
ZTE Corporation
No.50 Software Avenue, Yuhuatai District
Nanjing
China

Email: meng.wei2@zte.com.cn, vally.meng@gmail.com

Cui Wang
ZTE Corporation
No.50 Software Avenue, Yuhuatai District
Nanjing
China

Email: wang.cui1@zte.com.cn

