

SIDROPS Working Group
Internet Draft
Intended status: Standards Track
Expires: June 5, 2024

Y. Liu
China Mobile
C. Lin
New H3C Technologies
H. Wang
Huawei
H. Liu
ZTE
D. Ma
ZDNS
December 8, 2023

**YANG Data Model for RPKI to Router Protocol
draft-liu-sidrops-rtr-yang-04**

Abstract

This document defines YANG data models for configuring and managing Resource Public Key Infrastructure (RPKI) to Router Protocol ([RFC6810](#) and [RFC8210](#)).

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 June 5, 2024.

Copyright Notice

Copyright (c) 2023 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 Revised BSD License text as described in Section 4.e of the [Trust Legal Provisions](#) and are provided without warranty as described in the Revised BSD License.

Table of Contents

- [1. Introduction.....3](#)
- [1.1. Terminology.....3](#)
- [2. Model Overview.....3](#)
- [3. RPKI to Router YANG Module.....4](#)
- [3.1. Tree View.....4](#)
- [3.2. Yang Module.....8](#)
- [4. BGP Origin AS Validation YANG Module.....23](#)
- [4.1. Tree View.....23](#)
- [4.2. Yang Module.....26](#)
- [5. BGPsec YANG Module.....32](#)
- [5.1. Tree View.....32](#)
- [5.2. Yang Module.....35](#)
- [6. BGP ASPA YANG Module.....40](#)
- [6.1. Tree View.....40](#)
- [6.2. Yang Module.....42](#)
- [7. Security Considerations.....47](#)
- [8. IANA Considerations.....47](#)
- [9. References.....48](#)
- [9.1. Normative References.....48](#)
- [9.2. Informative References.....48](#)
- [Contributors.....49](#)
- [Authors' Addresses.....49](#)

1. Introduction

[RFC6810] and [RFC8210] describes a protocol to deliver Resource Public Key Infrastructure (RPKI) prefix origin data and router keys from a trusted cache server to a router, referred to as RPKI-Router protocol.

[RFC6811] validate the origination Autonomous System (AS) of BGP routes based on the Validated ROA Payload (VRP) received from the RPKI cache server.

This document defines YANG [RFC7950] data models for configuring and managing RPKI-Router Protocol ([RFC6810], [RFC8210], and [I-D.ietf-sidrops-8210bis]).

1.1. Terminology

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.

2. Model Overview

Three YANG data models are defined in this document.

The ietf-rpki-rtr.yang data model provides the methods for configuring and managing RPKI-Router Protocol. It includes:

- o Connectivity parameters, such as RPKI cache server IP address and destination port.
- o Session parameters, such as purge time, refresh time, response time, and the maximum number of received ROAs.
- o Session status and statistics, such as session ID, serial number, number of received and transmitted messages, and number of ROA records.
- o ROA records, router-key records, and ASPA records.

The ietf-bgp-origin-as-validation.yang data model provides the methods for configuring BGP origin AS validation.

- o Origin AS validation parameters for BGP routes.

- o Validity states of BGP routes.

The `ietf-bgp-sec.yang` data model provides the methods for configuring BGPsec.

- o BGPsec parameters for BGP routes.
- o Validity states of BGP routes.

[3. RPKI to Router YANG Module](#)

[3.1. Tree View](#)

The complete tree of the `ietf-rpki-rtr.yang` data model is represented as following. See [[RFC8340](#)] for an explanation of the symbols used.

```
module: ietf-rpki-rtr
```

```
+--rw rpki-rtr
  +--ro roa-table
    | +--ro ipv4
    | | +--ro roas
    | |   +--ro roa* [prefix max-len asn server-address]
    | |     +--ro prefix          inet:ipv4-prefix
    | |     +--ro max-len         ipv4-pfx-len
    | |     +--ro asn             inet:as-number
    | |     +--ro server-address  inet:ip-address
    | +--ro ipv6
    |   +--ro roas
    |     +--ro roa* [prefix max-len asn server-address]
    |       +--ro prefix          inet:ipv6-prefix
    |       +--ro max-len         ipv6-pfx-len
    |       +--ro asn             inet:as-number
    |       +--ro server-address  inet:ip-address
  +--ro router-key-table
    | +--ro router-keys
    |   +--ro router-key* [ski asn key server-address]
    |     +--ro ski              subject-key-id
    |     +--ro asn              inet:as-number
    |     +--ro key              string
    |     +--ro server-address    inet:ip-address
  +--ro aspa-table
    | +--ro ipv4
    | | +--ro aspas
    | |   +--ro aspa* [customer-asn server-address]
    | |     +--ro customer-asn    inet:as-number
    | |     +--ro server-address  inet:ip-address
    | |     +--ro provider-asns* [provider-asn]
    | |       +--ro provider-asn  inet:as-number
    | +--ro ipv6
    | | +--ro aspas
    | |   +--ro aspa* [customer-asn server-address]
    | |     +--ro customer-asn    inet:as-number
    | |     +--ro server-address  inet:ip-address
    | |     +--ro provider-asns* [provider-asn]
    | |       +--ro provider-asn  inet:as-number
  +--rw cache-servers
    +--rw cache-server* [server-address]
      +--rw server-address          inet:ip-address
      +--rw server-port?           inet:port-number
      +--rw local-address?         union
      +--rw local-port?           inet:port-number
      +--rw protocol-version?     uint32
      +--rw preference?           uint32
```



```
+--rw enabled?          boolean
+--rw description?     string
+--rw secure-session-enable? boolean
+--rw secure-session
| +--rw (option)?
|   +--:(md5)
|     | +--rw enable-md5?
|     | |         boolean
|     | +--rw md5-password?
|     | |         ianach:crypt-hash
|     +--:(ssh)
|       | +--rw enable-ssh?
|       | |         boolean
|       | +--rw ssh-client-grouping?
|       | |         identityref
|       +--:(keychain)
|         +--rw enable-keychain?
|         |         boolean
|         +--rw keychain-name?
|         |         key-chain:key-chain-ref
+--rw purge-time?      uint32
+--rw refresh-time?   uint32
+--rw response-time?  uint32
+--rw roa-limit
| +--rw max-number?          uint64
| +--rw threshold-percentage? uint8
| +--rw over-threshold-action? enumeration
| +--rw reconnect-interval?  uint32
+--rw aspa-limit
| +--rw max-number?          uint64
| +--rw threshold-percentage? uint8
| +--rw over-threshold-action? enumeration
| +--rw reconnect-interval?  uint32
+--ro session-state?  enumeration
+--ro session-id?     uint16
+--ro serial-number?  uint32
+--ro statistics
| +--ro in-total-messages?  yang:zero-based-counter64
| +--ro out-total-messages? yang:zero-based-counter64
| +--ro ipv4-roa-records?   yang:zero-based-counter64
| +--ro ipv6-roa-records?   yang:zero-based-counter64
| +--ro router-key-records? yang:zero-based-counter64
| +--ro ipv4-aspa-records?  yang:zero-based-counter64
| +--ro ipv6-aspa-records?  yang:zero-based-counter64
+--ro roa-table
| +--ro ipv4
| | +--ro roas
| | | +--ro roa* [prefix max-len asn]
```



```
| |      +--ro prefix  inet:ipv4-prefix
| |      +--ro max-len  ipv4-pfx-len
| |      +--ro asn      inet:as-number
| +--ro ipv6
|   +--ro roas
|     +--ro roa* [prefix max-len asn]
|       +--ro prefix  inet:ipv6-prefix
|       +--ro max-len  ipv6-pfx-len
|       +--ro asn      inet:as-number
+--ro router-key-table
| +--ro router-keys
|   +--ro router-key* [ski asn key]
|     +--ro ski          subject-key-id
|     +--ro asn          inet:as-number
|     +--ro key          string
+--ro aspa-table
  +--ro ipv4
  | +--ro aspas
  |   +--ro aspa* [customer-asn]
  |     +--ro customer-asn  inet:as-number
  |     +--ro provider-asns* [provider-asn]
  |       +--ro provider-asn inet:as-number
  +--ro ipv6
  | +--ro aspas
  |   +--ro aspa* [customer-asn]
  |     +--ro customer-asn  inet:as-number
  |     +--ro provider-asns* [provider-asn]
  |       +--ro provider-asn inet:as-number
```

3.2. Yang Module

```
<CODE BEGINS> file "ietf-rpki-rtr@2022-10-18.yang"

module ietf-rpki-rtr {
  yang-version "1.1";
  namespace "urn:ietf:params:xml:ns:yang:ietf-rpki-rtr";
  prefix "rpki-rtr";

  import ietf-yang-types {
    prefix "yang";
    reference
      "RFC 6991: Common YANG Data Types.";
  }

  import ietf-inet-types {
    prefix "inet";
    reference
      "RFC 6991: Common YANG Data Types";
  }

  import iana-crypt-hash {
    prefix "ianach";
    reference
      "RFC 7317: A YANG Data Model for System Management";
  }

  import ietf-ssh-client {
    prefix "ssh";
    reference
      "RFC XXXX: YANG Groupings for SSH Clients and SSH Servers";
  }

  import ietf-interfaces {
    prefix "if";
    reference
      "RFC 8343, A YANG Data Model for Interface Management.";
  }

  import ietf-key-chain {
    prefix key-chain;
    reference
      "RFC 8177: YANG Key Chain.";
  }

  organization
    "IETF SIDROPS Working Group";
```


contact

"TBD";

description

"This module describes a YANG model for the Resource Public Key Infrastructure (RPKI) to Router Protocol configuration.

This YANG model conforms to the Network Management Datastore Architecture (NMDA) as described in [RFC 8342](#).

Copyright (c) 2022 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.";

reference "RFC XXXX";

revision 2022-10-18 {

description

"Initial Version";

reference

"RFC XXXX, YANG Data Model for RPKI to Router Protocol";

}

typedef ipv4-pfx-len {

type uint8 {

range "0 .. 32";

}

description

"IPv4 Prefix Length.";

}

typedef ipv6-pfx-len {

type uint8 {


```
    range "0 .. 128";
  }
  description
    "IPv6 Prefix Length.";
}

typedef subject-key-id {
  type binary {
    length 20;
  }
  description
    "Subject Key Identifier.";
}

grouping records-limit {
  description
    "Limit of records that can be received from the RPKI
    cache server.";
  leaf max-number {
    type uint64;
    description
      "Configures the maximum number of ROAs that can be
      received from the RPKI cache server.";
  }
  leaf threshold-percentage {
    type uint8 {
      range "0..100";
    }
    units "percent";
    description
      "Configures the threshold percentage for ROA maximum
      number.";
  }
  leaf over-threshold-action {
    type enumeration {
      enum alert-only {
        description
          "Generates alert messages.";
      }
      enum discard {
        description
          "Discards excess ROAs.";
      }
      enum reconnect {
        description
          "Diconncets with the RPKI cache server,
          and tries to reconnect after reconnection
          timer expires.";
      }
    }
  }
}
```



```
    }
    enum idle-forever {
      description
        "Diconncets with the RPKI cache server
        forever.";
    }
  }
  description
    "The action to taken when ROA number exceeds
    threshold.";
}
leaf reconnect-interval {
  type uint32 {
    range "1..30000";
  }
  units "minutes";
  description
    "Time interval for the reconnection timer.";
}
}

grouping aspa-overall-records {
  description
    "ASPAs received from all RPKI cache servers.";
  list aspas {
    key "customer-asn";
    description
      "An entry of ASPA.";
    leaf customer-asn {
      type inet:as-number;
      description
        "The AS number of a customer.";
    }
    leaf server-address {
      type inet:ip-address;
      description
        "IP address of the RPKI cache server.";
    }
  }
  list provider-asns {
    key "provider-asn";
    description
      "Providers of the customer.";
    leaf provider-asn {
      type inet:as-number;
      description
        "The AS number of a provider.";
    }
  }
}
```



```
    }
  }

  grouping aspa-server-records {
    description
      "ASPA's received from a RPKI cache server.";
    list aspas {
      key "customer-asn";
      description
        "An entry of ASPA.";
      leaf customer-asn {
        type inet:as-number;
        description
          "The AS number of a customer.";
      }
      list provider-asns {
        key "provider-asn";
        description
          "Providers of the customer.";
        leaf provider-asn {
          type inet:as-number;
          description
            "The AS number of a provider.";
        }
      }
    }
  }
}

container rpki-rtr {
  description
    "Configuration parameters for the RPKI to Router Protocol.";
  container roa-table {
    config false;
    description
      "Table of ROAs received from all RPKI cache servers.";
    container ipv4 {
      config false;
      description
        "Container for IPV4 ROAs table.";
      container roas {
        config false;
        description
          "ROAs received from the RPKI cache server.";
        list roa {
          key "prefix max-len asn server-address";
          description
            "An entry of ROA.";
          leaf prefix {
```



```
    type inet:ipv4-prefix;
    description
      "The IPv4 prefix of the ROA.";
  }
  leaf max-len {
    type ipv4-pfx-len;
    description
      "Denotes the longest prefix allowed. This
      MUST NOT be less than the prefix length.";
  }
  leaf asn {
    type inet:as-number;
    description
      "The origin AS number of the ROA.";
  }
  leaf server-address {
    type inet:ip-address;
    description
      "IP address of the RPKI cache server.";
  }
}
}
}
}
container ipv6 {
  config false;
  description
    "Container for IPv6 ROAs table.";
  container roas {
    config false;
    description
      "ROAs received from the RPKI cache server.";
    list roa {
      key "prefix max-len asn server-address";
      description
        "An entry of ROA.";
      leaf prefix {
        type inet:ipv6-prefix;
        description
          "The IPv6 prefix of the ROA.";
      }
      leaf max-len {
        type ipv6-pfx-len;
        description
          "Denotes the longest prefix allowed. This
          MUST NOT be less than the prefix length.";
      }
      leaf asn {
        type inet:as-number;
```



```
        description
            "The origin AS number of the ROA.";
    }
    leaf server-address {
        type inet:ip-address;
        description
            "IP address of the RPKI cache server.";
    }
}
}
}
}
container router-key-table {
    config false;
    description
        "Table of router keys received from all RPKI cache
        servers.";
    container router-keys {
        config false;
        description
            "Router keys received from the RPKI cache server.";
        list router-key {
            key "ski asn key server-address";
            description
                "An entry of router key.";
            leaf ski {
                type subject-key-id;
                description
                    "A router key's Subject Key Identifier (SKI).";
                reference
                    "RFC 6487: A Profile for X.509 PKIX Resource
                    Certificates";
            }
            leaf asn {
                type inet:as-number;
                description
                    "The AS number of the router which the key
                    belongs to.";
            }
            leaf key {
                type string;
                description
                    "A router key's subjectPublicKeyInfo value.";
                reference
                    "RFC 8608: BGPsec Algorithms, Key Formats, and
                    Signature Formats";
            }
            leaf server-address {
```



```
        type inet:ip-address;
        description
            "IP address of the RPKI cache server.";
    }
}
}
}
container aspa-table {
    config false;
    description
        "Table of ASPAs received from all RPKI cache servers.";
    container ipv4 {
        config false;
        description
            "Container for IPv4 ASPAs table.";
        uses aspa-overall-records;
    }
    container ipv6 {
        config false;
        description
            "Container for IPv6 ASPAs table.";
        uses aspa-overall-records;
    }
}
container cache-servers {
    description
        "Parameters of RPKI cache servers.";
    list cache-server {
        key "server-address";
        description
            "Each entry contains parameters for a RPKI cache server
            identified by the 'server-address' key.";
        leaf server-address {
            type inet:ip-address;
            mandatory true;
            description
                "The IP address of the RPKI cache server";
        }
        leaf server-port {
            type inet:port-number;
            description
                "The remote port for the connection
                to the RPKI cache server";
        }
        leaf local-address {
            type union {
                type inet:ip-address;
                type if:interface-ref;
            }
        }
    }
}
```



```
    }
  description
    "The local IP (either IPv4 or IPv6) address to use for
    the connection to the RPKI cache server. This may be
    expressed as either an IP address or reference to the
    name of an interface.";
}
leaf local-port {
  type inet:port-number;
  description
    "The local port for the connection
    to the RPKI cache server";
}
leaf enabled {
  type boolean;
  default "true";
  description
    "Whether the RPKI cache server is enabled.";
}
leaf protocol-version {
  type uint32;
  description
    "The version number of the RPKI to Router Protocol.";
}
leaf preference {
  type uint32;
  description
    "The router's preference to connect to that cache.
    The lower the value, the more preferred.";
}
leaf description {
  type string;
  description
    "Textual description of the RPKI cache server";
}
leaf secure-session-enable {
  type boolean;
  default "false";
  description
    "Whether the session is secured.";
}
container secure-session {
  when "../secure-session-enable = 'true'";
  description
    "Container for describing how a particular session
    is to be secured.";
  choice option {
    description
```



```
    "Choice for session securing methods.";
  case md5 {
    leaf md5-password {
      type ianach:crypt-hash;
      description
        "The password for md5 authentication.";
    }
    description
      "Uses TCP-MD5 to secure the session.";
  }
  case ssh {
    uses ssh:ssh-client-grouping {
      reference
        "RFC XXXX: YANG Groupings for SSH Clients and
        SSH Servers";
    }
    description
      "Uses SSH to secure the session.";
  }
  case keychain {
    leaf keychain-name {
      type key-chain:key-chain-ref;
      description
        "Name of key chain.";
      reference
        "RFC 8177: YANG Key Chain.";
    }
    description
      "Uses key-chain to secure the session.";
  }
}
}
leaf purge-time {
  type uint32;
  description
    "Configures the time a router waits to keep data from
    the RPKI cache server after the session drops.";
}
leaf refresh-time {
  type uint32;
  description
    "Configures the time a router waits in between sending
    periodic serial queries to the RPKI cache server.";
}
leaf response-time {
  type uint32;
  description
    "Configures the time a router waits for a response
```



```
        after sending a serial or reset query to the RPKI
        cache server.";
    }
    container roa-limit {
        description
            "Limit of ROA records that can be received from the
            RPKI cache server.";
        uses records-limit;
    }
    container aspa-limit {
        description
            "Limit of ASPA records that can be received from the
            RPKI cache server.";
        uses records-limit;
    }
    leaf session-state {
        type enumeration {
            enum idle {
                description
                    "The session is down.";
            }
            enum connect {
                description
                    "The session is waiting for the underlying
                    transport session to be established.";
            }
            enum establish {
                description
                    "The session is up.";
            }
        }
        config false;
        description
            "The session state.";
    }
    leaf session-id {
        type uint16;
        config false;
        description
            "When a cache server is started, it generates a
            Session ID to identify the instance of the cache
            and to bind it to the sequence of Serial Numbers
            that cache instance will generate.";
        reference
            "RFC 6810, The Resource Public Key Infrastructure
            (RPKI) to Router Protocol
            RFC 8210, The Resource Public Key Infrastructure
            (RPKI) to Router Protocol, Version 1";
```



```
}
leaf serial-number {
  type uint32;
  config false;
  description
    "A 32-bit strictly increasing unsigned integer which
     wraps from 2^32-1 to 0. It denotes the logical
     version of a cache.";
  reference
    "RFC 6810, The Resource Public Key Infrastructure
     (RPKI) to Router Protocol
     RFC 8210, The Resource Public Key Infrastructure
     (RPKI) to Router Protocol, Version 1";
}
container statistics {
  config false;
  description
    "Statistics of the RPKI cache server.";
  leaf in-total-messages {
    type yang:zero-based-counter64;
    description
      "The total number of messages received from the
       RPKI cache server.";
  }
  leaf out-total-messages {
    type yang:zero-based-counter64;
    description
      "The total number of messages transmitted to the
       RPKI cache server.";
  }
  leaf ipv4-roa-records {
    type yang:zero-based-counter64;
    description
      "The number of ROAs for IPv4 prefixes received
       from the RPKI cache server.";
  }
  leaf ipv6-roa-records {
    type yang:zero-based-counter64;
    description
      "The number of ROAs for IPv6 prefixes received
       from the RPKI cache server.";
  }
  leaf router-key-records {
    type yang:zero-based-counter64;
    description
      "The number of router keys received from the RPKI
       cache server.";
  }
}
```



```
leaf ipv4-aspa-records {
  type yang:zero-based-counter64;
  description
    "The number of IPv4 ASPAs received from the RPKI
    cache server.";
}
leaf ipv6-aspa-records {
  type yang:zero-based-counter64;
  description
    "The number of IPv6 ASPAs received from the RPKI
    cache server.";
}
}
container roa-table {
  config false;
  description
    "Table of ROAs received from the RPKI cache server.";
  container ipv4 {
    config false;
    description
      "Container for IPv4 ROAs table.";
    container roas {
      config false;
      description
        "ROAs received from the RPKI cache server.";
      list roa {
        key "prefix max-len asn";
        description
          "An entry of ROA.";
        leaf prefix {
          type inet:ipv4-prefix;
          description
            "The IPv4 prefix of the ROA.";
        }
        leaf max-len {
          type ipv4-pfx-len;
          description
            "Denotes the longest prefix allowed. This
            MUST NOT be less than the prefix length.";
        }
        leaf asn {
          type inet:as-number;
          description
            "The origin AS number of the ROA.";
        }
      }
    }
  }
}
```



```
container ipv6 {
  config false;
  description
    "Container for IPv6 ROAs table.";
  container roas {
    config false;
    description
      "ROAs received from the RPKI cache server.";
    list roa {
      key "prefix max-len asn";
      description
        "An entry of ROA.";
      leaf prefix {
        type inet:ipv6-prefix;
        description
          "The IPv6 prefix of the ROA.";
      }
      leaf max-len {
        type ipv6-pfx-len;
        description
          "Denotes the longest prefix allowed. This
            MUST NOT be less than the prefix length.";
      }
      leaf asn {
        type inet:as-number;
        description
          "The origin AS number of the ROA.";
      }
    }
  }
}
reference
  "RFC 6810, The Resource Public Key Infrastructure
  (RPKI) to Router Protocol
  RFC 8210, The Resource Public Key Infrastructure
  (RPKI) to Router Protocol, Version 1";
}
container router-key-table {
  config false;
  description
    "Table of router keys received from all RPKI cache
    servers.";
  container router-keys {
    config false;
    description
      "Router keys received from the RPKI cache server.";
    list router-key {
      key "ski asn key";
    }
  }
}
```



```
description
  "An entry of router key.";
leaf ski {
  type subject-key-id;
  description
    "A router key's Subject Key Identifier (SKI).";
  reference
    "RFC 6487: A Profile for X.509 PKIX Resource
    Certificates";
}
leaf asn {
  type inet:as-number;
  description
    "The AS number of the router which the key
    belongs to.";
}
leaf key {
  type string;
  description
    "A router key's subjectPublicKeyInfo value.";
  reference
    "RFC 8608: BGPsec Algorithms, Key Formats, and
    Signature Formats";
}
}
}
}
container aspa-table {
  config false;
  description
    "Table of ASPAs received from the RPKI cache server.";
  container ipv4 {
    config false;
    description
      "Container for IPv4 ASPAs table.";
    uses aspa-server-records;
  }
  container ipv6 {
    config false;
    description
      "Container for IPv6 ASPAs table.";
    uses aspa-server-records;
  }
}
reference
  "RFC 6810, The Resource Public Key Infrastructure
  (RPKI) to Router Protocol
  RFC 8210, The Resource Public Key Infrastructure
  (RPKI) to Router Protocol, Version 1";
```



```
    }  
  }  
}
```

<CODE ENDS>

4. BGP Origin AS Validation YANG Module

4.1. Tree View

The complete tree of the `ietf-bgp-origin-as-validation.yang` data model is represented as following. See [[RFC8340](#)] for an explanation of the symbols used.

```
module: ietf-bgp-origin-as-validation
```

```
augment /rt:routing/rt:control-plane-protocols
  /rt:control-plane-protocol/bgp:bgp/bgp:global
  /bgp:afi-safis/bgp:afi-safi/bgp:ipv4-unicast:
  +--rw origin-as-validation
    +--rw enabled?          boolean
    +--rw redistribution-as? inet:as-number
```

```
augment /rt:routing/rt:control-plane-protocols
  /rt:control-plane-protocol/bgp:bgp/bgp:global
  /bgp:afi-safis/bgp:afi-safi/bgp:ipv6-unicast:
  +--rw origin-as-validation
    +--rw enabled?          boolean
    +--rw redistribution-as? inet:as-number
```

```
augment /rt:routing/rt:control-plane-protocols
  /rt:control-plane-protocol/bgp:bgp/bgp:rib
  /bgp:afi-safis/bgp:afi-safi/bgp:ipv4-unicast
  /bgp:loc-rib/bgp:routes/bgp:route:
  +--ro origin-as-validity? origin-as-validity-state
```

```
augment /rt:routing/rt:control-plane-protocols
  /rt:control-plane-protocol/bgp:bgp/bgp:rib
  /bgp:afi-safis/bgp:afi-safi/bgp:ipv6-unicast
  /bgp:loc-rib/bgp:routes/bgp:route:
  +--ro origin-as-validity? origin-as-validity-state
```

```
augment /rt:routing/rt:control-plane-protocols
  /rt:control-plane-protocol/bgp:bgp/bgp:global
  /bgp:afi-safis/bgp:afi-safi/bgp:route-selection-options:
  +--rw origin-as
    +--rw enabled?          boolean
    +--rw allow-invalid?    boolean
    +--rw allow-not-found?  boolean
```

```
augment /rt:routing/rt:control-plane-protocols
  /rt:control-plane-protocol/bgp:bgp/bgp:neighbors
  /bgp:neighbor/bgp:afi-safis/bgp:afi-safi
  /bgp:ipv4-unicast:
  +--rw send-origin-as-validity? boolean
  +--rw export-origin-as-validation
    +--rw enabled?          boolean
    +--rw allow-not-found?  boolean
```

```
augment /rt:routing/rt:control-plane-protocols
  /rt:control-plane-protocol/bgp:bgp/bgp:neighbors
  /bgp:neighbor/bgp:afi-safis/bgp:afi-safi
```



```
    /bgp:ipv6-unicast:
+--rw send-origin-as-validity?  boolean
+--rw export-origin-as-validation
    +--rw enabled?                boolean
    +--rw allow-not-found?        boolean

augment /rt:routing/rt:control-plane-protocols
    /rt:control-plane-protocol/bgp:bgp/bgp:peer-groups
    /bgp:peer-group/bgp:afi-safis/bgp:afi-safi
    /bgp:ipv4-unicast:
+--rw send-origin-as-validity?  boolean
+--rw export-origin-as-validation
    +--rw enabled?                boolean
    +--rw allow-not-found?        boolean

augment /rt:routing/rt:control-plane-protocols
    /rt:control-plane-protocol/bgp:bgp/bgp:peer-groups
    /bgp:peer-group/bgp:afi-safis/bgp:afi-safi
    /bgp:ipv6-unicast:
+--rw send-origin-as-validity?  boolean
+--rw export-origin-as-validation
    +--rw enabled?                boolean
    +--rw allow-not-found?        boolean
```

4.2. Yang Module

```
<CODE BEGINS> file "ietf-bgp-origin-as-validation@2022-10-18.yang"

module ietf-bgp-origin-as-validation {
  yang-version "1.1";
  namespace "urn:ietf:params:xml:ns:yang:"
    + "ietf-bgp-origin-as-validation";
  prefix "oav";

  import ietf-inet-types {
    prefix "inet";
    reference
      "RFC 6991: Common YANG Data Types";
  }

  import ietf-routing {
    prefix "rt";
    reference
      "RFC 8349, A YANG Data Model for Routing Management
      (NMDA Version).";
  }

  import ietf-bgp {
    prefix "bgp";
    reference
      "RFC XXXX: YANG Model for Border Gateway Protocol (BGP-4)";
  }

  import iana-bgp-types {
    prefix "bt";
    reference
      "RFC XXXX: YANG Model for Border Gateway Protocol (BGP-4)";
  }

  import iana-bgp-rib-types {
    prefix "brt";
    reference
      "RFC XXXX: YANG Model for Border Gateway Protocol (BGP-4)";
  }

  organization
    "IETF SIDROPS Working Group";

  contact
    "TBD";

  description
```


"This module describes configuration of the BGP origin AS validation.

This YANG model conforms to the Network Management Datastore Architecture (NMDA) as described in [RFC 8342](#).

Copyright (c) 2022 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.";

```
reference "RFC XXXX";
```

```
revision 2022-10-18 {  
  description  
    "Initial Version";  
  reference  
    "RFC XXXX, YANG Data Model for RPKI to Router Protocol";  
}
```

```
identity ineligible-orgen-as {  
  base brt:ineligible-route-reason;  
  description  
    "Route was ineligible due to origin as validation";  
}
```

```
typedef origin-as-validity-state {  
  type enumeration {  
    enum not-found {  
      description  
        "No VRP Covers the Route Prefix.";  
    }  
    enum valid {  
      description
```



```
        "At least one VRP Matches the Route Prefix.";
    }
    enum invalid {
        description
            "At least one VRP Covers the Route Prefix, but no VRP
            Matches it.";
    }
    enum disabled {
        description
            "BGP origin AS validation is not enabled.";
    }
}
description
    "Origin AS validation state of BGP routes.";
reference
    "RFC 6811, BGP Prefix Origin Validation.";
}

grouping origin-as-validation-config {
    description
        "Origin AS validation of BGP prefix.";
    container origin-as-validation {
        leaf enabled {
            type boolean;
            default "false";
            description
                "Whether origin-AS validation of BGP prefix is enabled.";
        }
        leaf redistribution-as {
            type inet:as-number;
            description
                "Uses this AS number in the origin-AS validation for
                redistributed routes since they have no AS-PATH.";
            reference
                "RFC 8481, Clarifications to BGP Origin Validation Based
                on Resource Public Key Infrastructure (RPKI).";
        }
    }
    description
        "Origin AS validation of BGP prefix.";
}

grouping origin-as-selection-option {
    description
        "Origin AS option for BGP route selection.";
    container origin-as {
        leaf enabled {
            type boolean;
```



```
        default "false";
        description
            "When enabled allows the origin AS validity states to be
            taken into consideration in the best-path calculation.";
    }
    leaf allow-invalid {
        type boolean;
        default "false";
        description
            "When enabled allows the route with 'invalid' origin AS
            to be taken into consideration in the best-path
            calculation.";
    }
    leaf allow-not-found {
        type boolean;
        default "true";
        description
            "When enabled allows the route with 'not-found' origin
            AS to be taken into consideration in the best-path
            calculation.";
    }
    description
        "Origin AS option for BGP route selection.";
}
}

grouping origin-as-validity-advertisement {
    description
        "Advertisement of Origin Validation State Extended
        Community to neighbor(s).";
    leaf send-origin-as-validity {
        type boolean;
        default "false";
        description
            "If set to true, send the origin AS validity to the
            neighbor(s) using Origin Validation State Extended
            Community";
        reference
            "RFC 8097, BGP Prefix Origin Validation State Extended
            Community.";
    }
}

grouping export-origin-as-validation-config {
    description
        "Export Origin AS validation of BGP prefix.";
    container export-origin-as-validation {
        leaf enabled {
```



```
    type boolean;
    default "false";
    description
      "When enabled allows the origin AS validity states to be
       taken into consideration in BGP export.";
  }
  leaf allow-not-found {
    type boolean;
    default "false";
    description
      "When enabled allows the route with 'not-found' origin
       AS to be sent to the neighbor.";
  }
  description
    "Export Origin AS validation of BGP prefix.";
  reference
    "RFC 8893, Resource Public Key Infrastructure (RPKI) Origin
     Validation for BGP Export.";
}
}

augment "/rt:routing/rt:control-plane-protocols"
  + "/rt:control-plane-protocol/bgp:bgp/bgp:global"
  + "/bgp:afi-safis/bgp:afi-safi/bgp:ipv4-unicast" {
  description
    "Origin AS validation augmentation of BGP IPv4 Unicast
     Address Family.";
  uses origin-as-validation-config;
}

augment "/rt:routing/rt:control-plane-protocols"
  + "/rt:control-plane-protocol/bgp:bgp/bgp:global"
  + "/bgp:afi-safis/bgp:afi-safi/bgp:ipv6-unicast" {
  description
    "Origin AS validation augmentation of BGP IPv6 Unicast
     Address Family.";
  uses origin-as-validation-config;
}

augment "/rt:routing/rt:control-plane-protocols"
  + "/rt:control-plane-protocol/bgp:bgp/bgp:rib"
  + "/bgp:afi-safis/bgp:afi-safi/bgp:ipv4-unicast"
  + "/bgp:loc-rib/bgp:routes/bgp:route" {
  description
    "Origin AS validity augmentation of BGP IPv4 Unicast
     route.";
  leaf origin-as-validity {
    type origin-as-validity-state;
  }
}
```



```
        description
            "Origin AS validity of BGP IPv4 Unicast prefix";
    }
}

augment "/rt:routing/rt:control-plane-protocols"
    + "/rt:control-plane-protocol/bgp:bgp/bgp:rib"
    + "/bgp:afi-safis/bgp:afi-safi/bgp:ipv6-unicast"
    + "/bgp:loc-rib/bgp:routes/bgp:route" {
    description
        "Origin AS validity augmentation of BGP IPv6 Unicast
        route.";
    leaf origin-as-validity {
        type origin-as-validity-state;
        description
            "Origin AS validity of BGP IPv6 Unicast prefix";
    }
}

augment "/rt:routing/rt:control-plane-protocols"
    + "/rt:control-plane-protocol/bgp:bgp/bgp:global"
    + "/bgp:afi-safis/bgp:afi-safi"
    + "/bgp:route-selection-options" {
    when "derived-from-or-self(..../bgp:name, 'bt:ipv4-unicast') or "
        + "derived-from-or-self(..../bgp:name, 'bt:ipv6-unicast')" {
        description
            "This augmentation is valid for IPv4 and IPv6 Unicast.";
    }
    description
        "augmentation of BGP route selection options";
    uses origin-as-selection-option;
}

augment "/rt:routing/rt:control-plane-protocols"
    + "/rt:control-plane-protocol/bgp:bgp/bgp:neighbors"
    + "/bgp:neighbor/bgp:afi-safis/bgp:afi-safi"
    + "/bgp:ipv4-unicast" {
    description
        "augmentation of Origin Validation State Extended
        Community advertisement for IPv4 Unicast neighbor";
    uses origin-as-validity-advertisement;
    uses export-origin-as-validation-config;
}

augment "/rt:routing/rt:control-plane-protocols"
    + "/rt:control-plane-protocol/bgp:bgp/bgp:neighbors"
    + "/bgp:neighbor/bgp:afi-safis/bgp:afi-safi"
    + "/bgp:ipv6-unicast" {
```



```
    description
      "augmentation of Origin Validation State Extended
      Community advertisement for IPv6 Unicast neighbor";
    uses origin-as-validity-advertisement;
    uses export-origin-as-validation-config;
  }

  augment "/rt:routing/rt:control-plane-protocols"
    + "/rt:control-plane-protocol/bgp:bgp/bgp:peer-groups"
    + "/bgp:peer-group/bgp:afi-safis/bgp:afi-safi"
    + "/bgp:ipv4-unicast" {
    description
      "augmentation of Origin Validation State Extended
      Community advertisement for IPv4 Unicast peer group";
    uses origin-as-validity-advertisement;
    uses export-origin-as-validation-config;
  }

  augment "/rt:routing/rt:control-plane-protocols"
    + "/rt:control-plane-protocol/bgp:bgp/bgp:peer-groups"
    + "/bgp:peer-group/bgp:afi-safis/bgp:afi-safi"
    + "/bgp:ipv6-unicast" {
    description
      "augmentation of Origin Validation State Extended
      Community advertisement for IPv6 Unicast peer group";
    uses origin-as-validity-advertisement;
    uses export-origin-as-validation-config;
  }
}

<CODE ENDS>
```

5. BGPsec YANG Module

5.1. Tree View

The complete tree of the `ietf-bgp-sec.yang` data model is represented as following. See [[RFC8340](#)] for an explanation of the symbols used.

```
module: ietf-bgp-sec
```

```
augment /rt:routing/rt:control-plane-protocols
  /rt:control-plane-protocol/bgp:bgp/bgp:global
  /bgp:afi-safis/bgp:afi-safi/bgp:ipv4-unicast:
  +--rw bgpsec-validation
    +--rw enabled?          boolean
```

```
augment /rt:routing/rt:control-plane-protocols
  /rt:control-plane-protocol/bgp:bgp/bgp:global
  /bgp:afi-safis/bgp:afi-safi/bgp:ipv6-unicast:
  +--rw bgpsec-validation
    +--rw enabled?          boolean
```

```
augment /rt:routing/rt:control-plane-protocols
  /rt:control-plane-protocol/bgp:bgp/bgp:rib
  /bgp:afi-safis/bgp:afi-safi/bgp:ipv4-unicast
  /bgp:loc-rib/bgp:routes/bgp:route:
  +--ro bgpsec-validity?    bgpsec-validity-state
```

```
augment /rt:routing/rt:control-plane-protocols
  /rt:control-plane-protocol/bgp:bgp/bgp:rib
  /bgp:afi-safis/bgp:afi-safi/bgp:ipv6-unicast
  /bgp:loc-rib/bgp:routes/bgp:route:
  +--ro bgpsec-validity?    bgpsec-validity-state
```

```
augment /rt:routing/rt:control-plane-protocols
  /rt:control-plane-protocol/bgp:bgp/bgp:global
  /bgp:afi-safis/bgp:afi-safi/bgp:route-selection-options:
  +--rw bgpsec
    +--rw enabled?          boolean
    +--rw allow-invalid?    boolean
```

```
augment /rt:routing/rt:control-plane-protocols
  /rt:control-plane-protocol/bgp:bgp/bgp:neighbors
  /bgp:neighbor/bgp:afi-safis/bgp:afi-safi
  /bgp:ipv4-unicast:
  +--rw export-bgpsec-validation
    +--rw enabled?          boolean
    +--rw allow-invalid?    boolean
```

```
augment /rt:routing/rt:control-plane-protocols
  /rt:control-plane-protocol/bgp:bgp/bgp:neighbors
  /bgp:neighbor/bgp:afi-safis/bgp:afi-safi
  /bgp:ipv6-unicast:
  +--rw export-bgpsec-validation
    +--rw enabled?          boolean
```



```
augment /rt:routing/rt:control-plane-protocols
  /rt:control-plane-protocol/bgp:bgp/bgp:peer-groups
  /bgp:peer-group/bgp:afi-safis/bgp:afi-safi
  /bgp:ipv4-unicast:
  +-rw export-bgpsec-validation
  +-rw enabled?          boolean

augment /rt:routing/rt:control-plane-protocols
  /rt:control-plane-protocol/bgp:bgp/bgp:peer-groups
  /bgp:peer-group/bgp:afi-safis/bgp:afi-safi
  /bgp:ipv6-unicast:
  +-rw export-bgpsec-validation
  +-rw enabled?          boolean
```

5.2. Yang Module

```
<CODE BEGINS> file "ietf-bgp-sec@2022-10-18.yang"

module ietf-bgp-sec {
  yang-version "1.1";
  namespace "urn:ietf:params:xml:ns:yang:"
    + "ietf-bgp-sec";
  prefix "oav";

  import ietf-routing {
    prefix "rt";
    reference
      "RFC 8349, A YANG Data Model for Routing Management
      (NMDA Version).";
  }

  import ietf-bgp {
    prefix "bgp";
    reference
      "RFC XXXX: YANG Model for Border Gateway Protocol (BGP-4)";
  }

  import iana-bgp-types {
    prefix "bt";
    reference
      "RFC XXXX: YANG Model for Border Gateway Protocol (BGP-4)";
  }

  import iana-bgp-rib-types {
    prefix "brt";
    reference
      "RFC XXXX: YANG Model for Border Gateway Protocol (BGP-4)";
  }

  organization
    "IETF SIDROPS Working Group";

  contact
    "TBD";

  description
    "This module describes configuration of BGPsec.

    This YANG model conforms to the Network Management
    Datastore Architecture (NMDA) as described in RFC 8342.

    Copyright (c) 2022 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.";

```
reference "RFC XXXX";

revision 2022-10-18 {
  description
    "Initial Version";
  reference
    "RFC XXXX, YANG Data Model for RPKI to Router Protocol";
}

identity ineligible-bgp {
  base brt:ineligible-route-reason;
  description
    "Route was ineligible due to bgpsec";
}

typedef bgpsec-validity-state {
  type enumeration {
    enum valid {
      description
        "The BGPsec UPDATE message is valid.";
    }
    enum invalid {
      description
        "The BGPsec UPDATE message is invalid.";
    }
    enum disabled {
      description
        "BGPsec validation is not enabled.";
    }
  }
}
```



```
    description
      "BGPsec validation state of BGP routes.";
    reference
      "RFC 8205, BGPsec Protocol Specification.";
  }

grouping bgpsec-validation-config {
  description
    "BGPsec validation of BGP prefix.";
  container bgpsec-validation {
    leaf enabled {
      type boolean;
      default "false";
      description
        "Whether BGPsec validation of BGP prefix is enabled.";
    }
    description
      "BGPsec validation of BGP prefix.";
  }
}

grouping bgpsec-selection-option {
  description
    "BGPsec option for BGP route selection.";
  container bgpsec {
    leaf enabled {
      type boolean;
      default "false";
      description
        "When enabled allows the BGPsec validity states to be
        taken into consideration in the best-path calculation.";
    }
    leaf allow-invalid {
      type boolean;
      default "false";
      description
        "When enabled allows the route with 'invalid' BGPsec
        to be taken into consideration in the best-path
        calculation.";
    }
    description
      "BGPsec option for BGP route selection.";
  }
}

grouping export-bgpsec-validation-config {
  description
    "Export BGPsec validation of BGP prefix.";
```



```
    container export-bgpsec-validation {
      leaf enabled {
        type boolean;
        default "false";
        description
          "When enabled allows the BGPsec validity states to be
           taken into consideration in BGP export.";
      }
      description
        "Export BGPsec validation of BGP prefix.";
    }
  }
}

augment "/rt:routing/rt:control-plane-protocols"
  + "/rt:control-plane-protocol/bgp:bgp/bgp:global"
  + "/bgp:afi-safis/bgp:afi-safi/bgp:ipv4-unicast" {
  description
    "BGPsec augmentation of BGP IPv4 Unicast Address Family.";
  uses bgpsec-validation-config;
}

augment "/rt:routing/rt:control-plane-protocols"
  + "/rt:control-plane-protocol/bgp:bgp/bgp:global"
  + "/bgp:afi-safis/bgp:afi-safi/bgp:ipv6-unicast" {
  description
    " BGPsec augmentation of BGP IPv6 Unicast Address Family.";
  uses bgpsec-validation-config;
}

augment "/rt:routing/rt:control-plane-protocols"
  + "/rt:control-plane-protocol/bgp:bgp/bgp:rib"
  + "/bgp:afi-safis/bgp:afi-safi/bgp:ipv4-unicast"
  + "/bgp:loc-rib/bgp:routes/bgp:route" {
  description
    " BGPsec augmentation of BGP IPv4 Unicast route.";
  leaf bgpsec-validity {
    type bgpsec-validity-state;
    description
      "BGPsec validity of BGP IPv4 Unicast prefix";
  }
}

augment "/rt:routing/rt:control-plane-protocols"
  + "/rt:control-plane-protocol/bgp:bgp/bgp:rib"
  + "/bgp:afi-safis/bgp:afi-safi/bgp:ipv6-unicast"
  + "/bgp:loc-rib/bgp:routes/bgp:route" {
  description
    "BGPsec augmentation of BGP IPv6 Unicast route.";
```



```
    leaf bgpsec-validity {
      type bgpsec-validity-state;
      description
        "BGPsec validity of BGP IPv6 Unicast prefix";
    }
  }

augment "/rt:routing/rt:control-plane-protocols"
  + "/rt:control-plane-protocol/bgp:bgp/bgp:global"
  + "/bgp:afi-safis/bgp:afi-safi"
  + "/bgp:route-selection-options" {
  when "derived-from-or-self(..../bgp:name, 'bt:ipv4-unicast') or "
    + "derived-from-or-self(..../bgp:name, 'bt:ipv6-unicast')" {
    description
      "This augmentation is valid for IPv4 and IPv6 Unicast.";
  }
  description
    "augmentation of BGP route selection options";
  uses bgpsec-selection-option;
}

augment "/rt:routing/rt:control-plane-protocols"
  + "/rt:control-plane-protocol/bgp:bgp/bgp:neighbors"
  + "/bgp:neighbor/bgp:afi-safis/bgp:afi-safi"
  + "/bgp:ipv4-unicast" {
  description
    "augmentation of BGPSec for IPv4 Unicast neighbor";
  uses export-bgpsec-validation-config;
}

augment "/rt:routing/rt:control-plane-protocols"
  + "/rt:control-plane-protocol/bgp:bgp/bgp:neighbors"
  + "/bgp:neighbor/bgp:afi-safis/bgp:afi-safi"
  + "/bgp:ipv6-unicast" {
  description
    "augmentation of BGPSec for IPv6 Unicast neighbor";
  uses export-bgpsec-validation-config;
}

augment "/rt:routing/rt:control-plane-protocols"
  + "/rt:control-plane-protocol/bgp:bgp/bgp:peer-groups"
  + "/bgp:peer-group/bgp:afi-safis/bgp:afi-safi"
  + "/bgp:ipv4-unicast" {
  description
    "augmentation of BGPSec for IPv4 Unicast peer group";
  uses export-bgpsec-validation-config;
}
```



```
augment "/rt:routing/rt:control-plane-protocols"  
  + "/rt:control-plane-protocol/bgp:bgp/bgp:peer-groups"  
  + "/bgp:peer-group/bgp:afi-safis/bgp:afi-safi"  
  + "/bgp:ipv6-unicast" {  
    description  
      "augmentation of BGPsec for IPv6 Unicast peer group";  
    uses export-bgpsec-validation-config;  
  }  
}
```

<CODE ENDS>

6. BGP ASPA YANG Module

6.1. Tree View

The complete tree of the `ietf-bgp-aspa.yang` data model is represented as following. See [[RFC8340](#)] for an explanation of the symbols used.

```
module: ietf-bgp-aspa
```

```
augment /rt:routing/rt:control-plane-protocols
  /rt:control-plane-protocol/bgp:bgp/bgp:neighbors
  /bgp:neighbor:
  +--rw peer-role?          peer-role

augment /rt:routing/rt:control-plane-protocols
  /rt:control-plane-protocol/bgp:bgp/bgp:peer-groups
  /bgp:peer-group:
  +--rw peer-role?          peer-role

augment /rt:routing/rt:control-plane-protocols
  /rt:control-plane-protocol/bgp:bgp/bgp:global
  /bgp:afi-safis/bgp:afi-safi/bgp:ipv4-unicast:
  +--rw aspa-verification
  +--rw enabled?            boolean

augment /rt:routing/rt:control-plane-protocols
  /rt:control-plane-protocol/bgp:bgp/bgp:global
  /bgp:afi-safis/bgp:afi-safi/bgp:ipv6-unicast:
  +--rw aspa-verification
  +--rw enabled?            boolean

augment /rt:routing/rt:control-plane-protocols
  /rt:control-plane-protocol/bgp:bgp/bgp:rib
  /bgp:afi-safis/bgp:afi-safi/bgp:ipv4-unicast
  /bgp:loc-rib/bgp:routes/bgp:route:
  +--ro aspa-verification-state? aspa-verification-state

augment /rt:routing/rt:control-plane-protocols
  /rt:control-plane-protocol/bgp:bgp/bgp:rib
  /bgp:afi-safis/bgp:afi-safi/bgp:ipv6-unicast
  /bgp:loc-rib/bgp:routes/bgp:route:
  +--ro aspa-verification-state? aspa-verification-state

augment /rt:routing/rt:control-plane-protocols
  /rt:control-plane-protocol/bgp:bgp/bgp:global
  /bgp:afi-safis/bgp:afi-safi/bgp:route-selection-options:
  +--rw aspa
  +--rw enabled?            boolean
  +--rw allow-invalid?      boolean
  +--rw allow-unknown?      boolean
```


6.2. Yang Module

```
<CODE BEGINS> file "ietf-bgp-aspa@2022-10-18.yang"

module ietf-bgp-aspa {
  yang-version "1.1";
  namespace "urn:ietf:params:xml:ns:yang:"
    + "ietf-bgp-aspa";
  prefix "oav";

  import ietf-routing {
    prefix "rt";
    reference
      "RFC 8349, A YANG Data Model for Routing Management
      (NMDA Version).";
  }

  import ietf-bgp {
    prefix "bgp";
    reference
      "RFC XXXX: YANG Model for Border Gateway Protocol (BGP-4)";
  }

  import iana-bgp-types {
    prefix "bt";
    reference
      "RFC XXXX: YANG Model for Border Gateway Protocol (BGP-4)";
  }

  import iana-bgp-rib-types {
    prefix "brt";
    reference
      "RFC XXXX: YANG Model for Border Gateway Protocol (BGP-4)";
  }

  organization
    "IETF SIDROPS Working Group";

  contact
    "TBD";

  description
    "This module describes configuration of the BGP AS PATH
    Verification Based on ASPA.

    This YANG model conforms to the Network Management
    Datastore Architecture (NMDA) as described in RFC 8342."
  }
}
```


Copyright (c) 2022 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.";

```
reference "RFC XXXX";

revision 2022-10-18 {
  description
    "Initial Version";
  reference
    "RFC XXXX, YANG Data Model for RPKI to Router Protocol";
}

identity ineligible-aspa {
  base brt:ineligible-route-reason;
  description
    "Route was ineligible due to ASPA verification";
}

typedef peer-role {
  type enumeration {
    enum customer {
      description
        "The role of the BGP peer is customer.";
    }
    enum provider {
      description
        "The role of the BGP peer is provider.";
    }
  }
  enum lateral-peer {
    description
      "The role of the BGP peer is lateral peer.";
  }
}
```



```
enum rs {
  description
    "The role of the BGP peer is Route Server (RS).";
}
enum rs-client {
  description
    "The role of the BGP peer is RS-client.";
}
enum mutual-transit {
  description
    "The role of the BGP peer is mutual-transit.";
}
}
description
  "Roles of BGP peers.";
reference
  "RFC XXXX, BGP AS_PATH Verification Based on Autonomous
  System Provider Authorization (ASPA) Objects.";
}

typedef aspa-verification-state {
  type enumeration {
    enum valid {
      description
        "The ASPA verification outcome is valid.";
    }
    enum invalid {
      description
        "The ASPA verification outcome is invalid.";
    }
    enum unknown {
      description
        "The ASPA verification outcome is unknown.";
    }
    enum disabled {
      description
        "BGP ASPA verification is not enabled.";
    }
  }
}
description
  "ASPA verification state of BGP routes.";
reference
  "RFC XXXX, BGP AS_PATH Verification Based on Autonomous
  System Provider Authorization (ASPA) Objects.";
}

grouping aspa-config {
  description
```



```
    "ASPA verification of BGP prefix.";
  container aspa-verification {
    leaf enabled {
      type boolean;
      default "false";
      description
        "Whether ASPA verification of BGP prefix is enabled.";
    }
    description
      "ASPA verification of BGP prefix.";
  }
}

grouping aspa-selection-option {
  description
    "ASPA option for BGP route selection.";
  container aspa {
    leaf enabled {
      type boolean;
      default "false";
      description
        "When enabled allows the ASPA verification states to be
        taken into consideration in the best-path calculation.";
    }
    leaf allow-invalid {
      type boolean;
      default "false";
      description
        "When enabled allows the route with 'invalid' ASPA
        verification state to be taken into consideration in
        the best-path calculation.";
    }
    leaf allow-unknown {
      type boolean;
      default "true";
      description
        "When enabled allows the route with 'unknown' ASPA
        verification state to be taken into consideration in
        the best-path calculation.";
    }
  }
  description
    "ASPA option for BGP route selection.";
}

augment "/rt:routing/rt:control-plane-protocols"
  + "/rt:control-plane-protocol/bgp:bgp/bgpb:neighbors"
  + "/bgp:neighbor" {
```



```
description
  "augmentation of BGP peer roles for neighbors";
leaf peer-role {
  type peer-role;
  description
    "Role of the peer";
}
}

augment "/rt:routing/rt:control-plane-protocols"
  + "/rt:control-plane-protocol/bgp:bgp/bgp:peer-groups"
  + "/bgp:peer-group" {
  description
    "augmentation of BGP peer roles for peer groups";
  leaf peer-role {
    type peer-role;
    description
      "Role of the peer group";
  }
}

augment "/rt:routing/rt:control-plane-protocols"
  + "/rt:control-plane-protocol/bgp:bgp/bgp:global"
  + "/bgp:afi-safis/bgp:afi-safi/bgp:ipv4-unicast" {
  description
    "ASPA verification augmentation of BGP IPv4 Unicast
    Address Family.";
  uses aspa-config;
}

augment "/rt:routing/rt:control-plane-protocols"
  + "/rt:control-plane-protocol/bgp:bgp/bgp:global"
  + "/bgp:afi-safis/bgp:afi-safi/bgp:ipv6-unicast" {
  description
    "ASPA verification augmentation of BGP IPv6 Unicast
    Address Family.";
  uses aspa-config;
}

augment "/rt:routing/rt:control-plane-protocols"
  + "/rt:control-plane-protocol/bgp:bgp/bgp:rib"
  + "/bgp:afi-safis/bgp:afi-safi/bgp:ipv4-unicast"
  + "/bgp:loc-rib/bgp:routes/bgp:route" {
  description
    "ASPA verification state augmentation of BGP IPv4
    Unicast route.";
  leaf aspa-verification-state {
    type aspa-verification-state;
  }
}
```



```
        description
            "ASPA verification state of BGP IPv4 Unicast prefix.";
    }
}

augment "/rt:routing/rt:control-plane-protocols"
    + "/rt:control-plane-protocol/bgp:bgp/bgp:rib"
    + "/bgp:afi-safis/bgp:afi-safi/bgp:ipv6-unicast"
    + "/bgp:loc-rib/bgp:routes/bgp:route" {
    description
        "ASPA verification state augmentation of BGP IPv6
        Unicast route.";
    leaf aspa-verification-state {
        type aspa-verification-state;
        description
            "ASPA verification state of BGP IPv6 Unicast prefix.";
    }
}

augment "/rt:routing/rt:control-plane-protocols"
    + "/rt:control-plane-protocol/bgp:bgp/bgp:global"
    + "/bgp:afi-safis/bgp:afi-safi"
    + "/bgp:route-selection-options" {
    when "derived-from-or-self(..../bgp:name, 'bt:ipv4-unicast') or "
        + "derived-from-or-self(..../bgp:name, 'bt:ipv6-unicast')" {
        description
            "This augmentation is valid for IPv4 and IPv6 Unicast.";
    }
    description
        "augmentation of BGP route selection options";
    uses aspa-selection-option;
}
}
```

<CODE ENDS>

[7. Security Considerations](#)

TBD

[8. IANA Considerations](#)

TBD

9. References

9.1. 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>>.
- [RFC6810] Bush, R. and R. Austein, "The Resource Public Key Infrastructure (RPKI) to Router Protocol", [RFC 6810](#), DOI 10.17487/RFC6810, January 2013, <<https://www.rfc-editor.org/info/rfc6810>>.
- [RFC6811] Mohapatra, P., Scudder, J., Ward, D., Bush, R., and R. Austein, "BGP Prefix Origin Validation", [RFC 6811](#), DOI 10.17487/RFC6811, January 2013, <<https://www.rfc-editor.org/info/rfc6811>>.
- [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>>.
- [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>>.
- [RFC8210] Bush, R. and R. Austein, "The Resource Public Key Infrastructure (RPKI) to Router Protocol, Version 1", [RFC 8210](#), DOI 10.17487/RFC8210, September 2017, <<https://www.rfc-editor.org/info/rfc8210>>.
- [I-D.ietf-sidrops-8210bis] Bush, R. and R. Austein, "The Resource Public Key Infrastructure (RPKI) to Router Protocol, Version 2", Work in Progress, Internet-Draft, [draft-ietf-sidrops-8210bis-11](#), 21 September 2023, <<https://datatracker.ietf.org/doc/html/draft-ietf-sidrops-8210bis-11>>.

9.2. Informative References

- [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>>.

Contributors

Mengxiao Chen
H3C
China
Email: chen.mengxiao@h3c.com

Authors' Addresses

Yisong Liu
China Mobile
China
Email: liuyisong@chinamobile.com

Changwang Lin
New H3C Technologies
China
Email: linchangwang.04414@h3c.com

Haibo Wang
Huawei Technologies
China
Email: rainsword.wang@huawei.com

Hongwei Liu
ZTE Corporation
China
Email: liu.hongwei3@zte.com.cn

Di Ma
ZDNS
Floor 21, Block B, Greenland Center
Chaoyang Beijing, 100102
China
Email: madi@zdns.cn