A YANG Model for SAVNET

draft-li-savnet-sav-yang

@IETF 117 SAVNET WG

Dan Li (Tsinghua University)
Fang Gao (ZGC Lab)
Changwang Lin (New H3C)
Jianping Wu (Tsinghua University)
Zhen Tan (Huawei)
Weiqiang Cheng (China Mobile)

1

SAVNET YANG

- This Yang model provides a base framework for configuring and managing an SAV subsystem, including SAV table and SAV rules, and also specifies the model for the static SAV rules application.

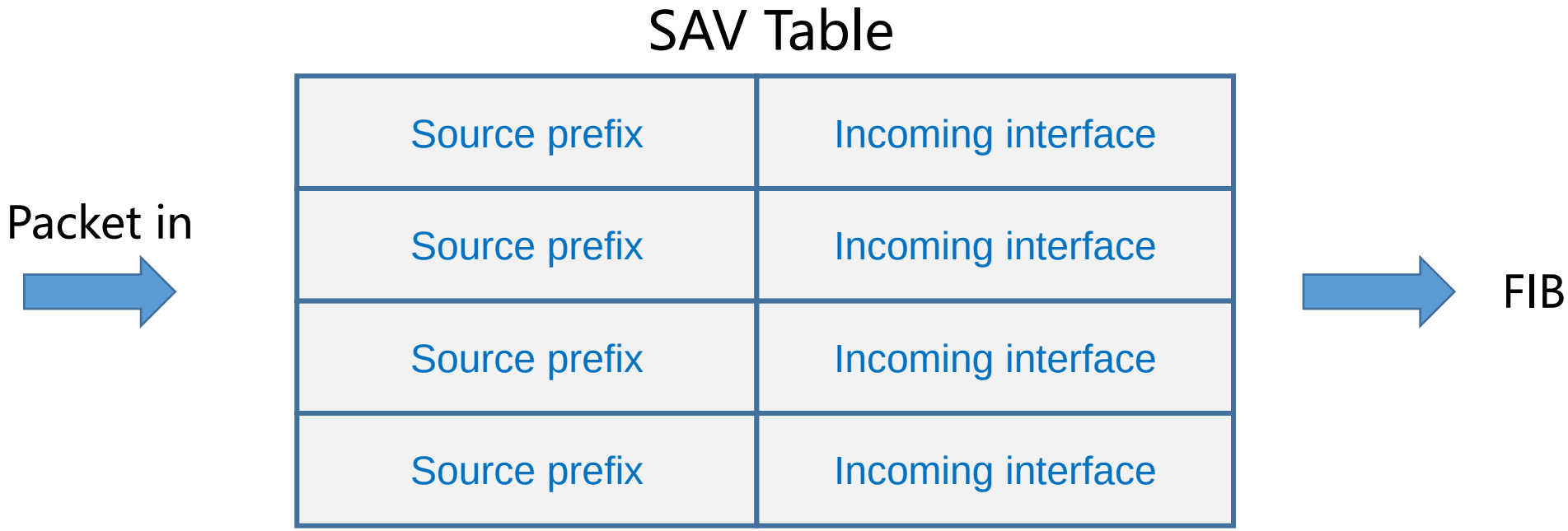
- The data model is designed by referring to the SAVNET functions defined in related drafts. [draft-li-savnet-intra-domain-architecture, draft-wu-savnet-inter-domain-architecture, draft-huang-savnet-sav-table]

- The model includes 3 parts: SAVNET configuration, state information, event notifications.

SAVNET YANG-02 Version

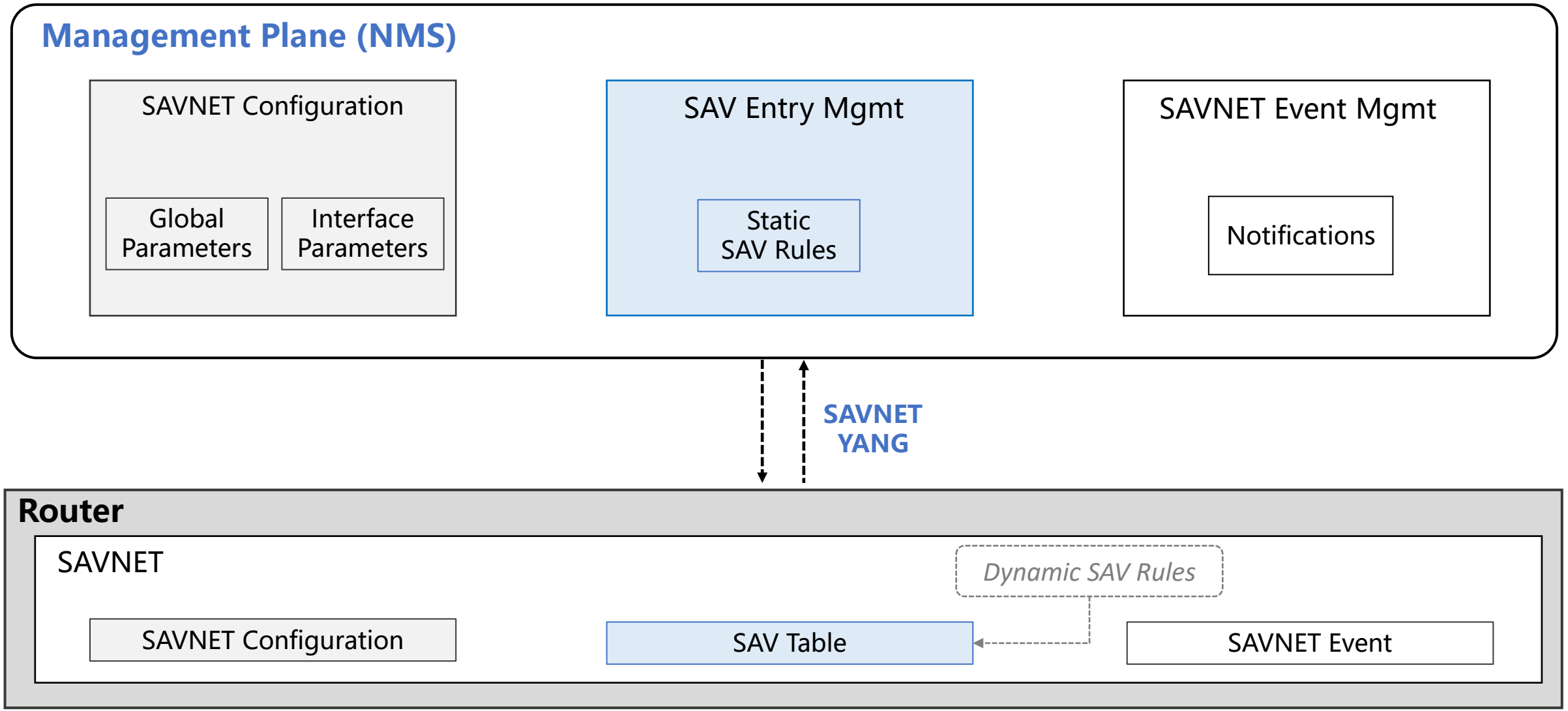
- The data model is designed based on implementation experience and Yang rules.
- The 02-Version is an initial version (and not the final version). It will be updated continuously according to the comments.
- Edited in compliance with RFC8407.

Building Block: SAV Table and SAV Rules

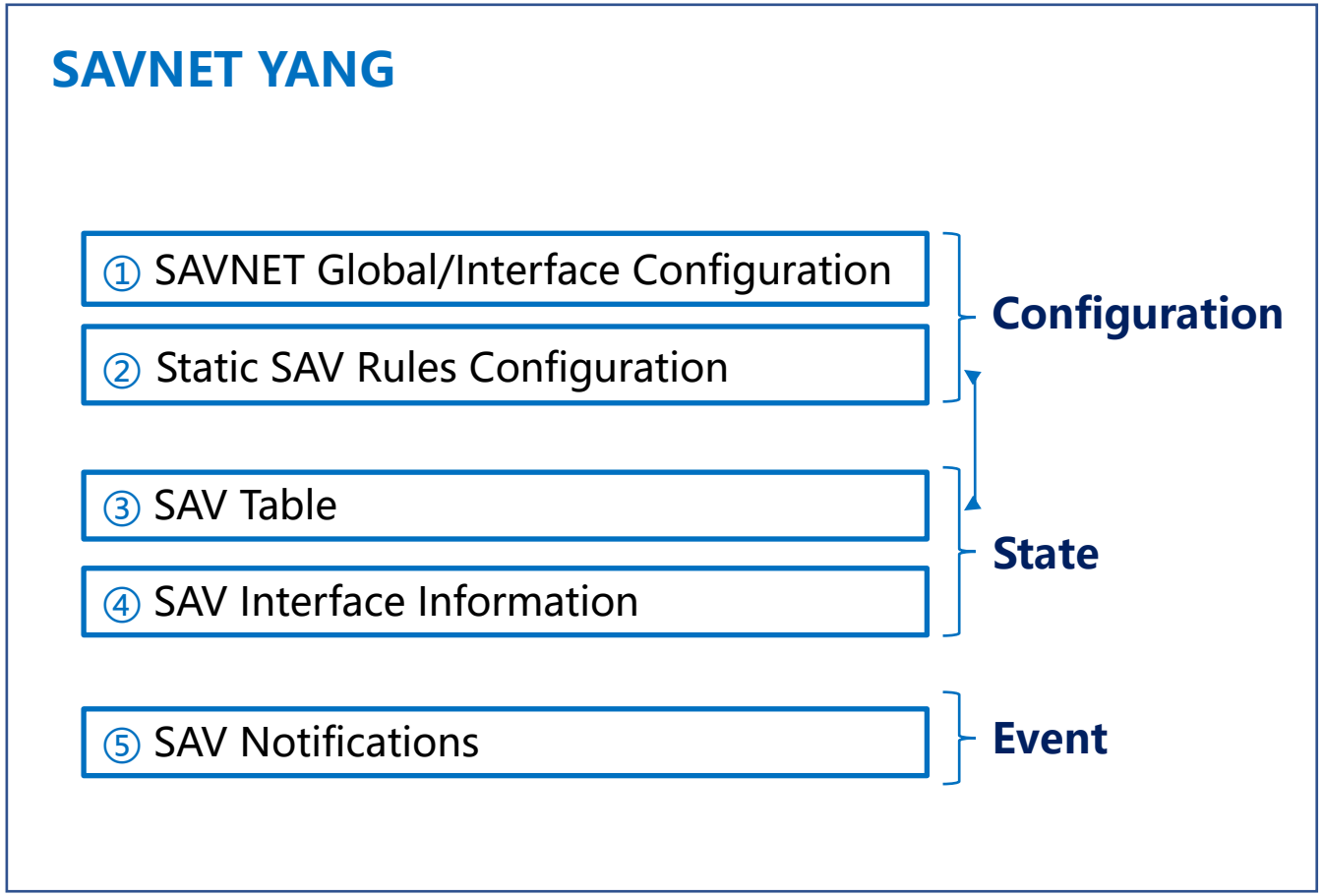


- Source address validation (SAV) table is constituted by a list of SAV rules. The table will take effect in data plane for checking the validity of source addresses.
- The Yang model makes a reference to RFC8349 [Yang model for RIB].

Operation of SAVNET Feature



Tree Diagram Overview




```
module: ietf-sav
   +--rw sav
   |  +--rw router-id?  yang:dotted-quad ①
   |  +--rw v4sav-entry-limits
   |  +--rw sav-controls
   |  |     ...
augment /if:interfaces/if:interface: ①
   |  +--rw sav-control
   |  |  +--rw sav-enabled? boolean
   |  |        ...
   |  +--rw static-savs ②
   |  |  +--rw v4sav:ipv4
   |  |  |     ...
   |  |  +--rw v6sav:ipv6
   |  |        ...
   |  +--rw sav-tables ③
   |  |  +--rw sav-table* [name]
   |  |     +--ro sav-rules
   |  |        ...
   |  +--ro interfaces ④
   |  |  +--ro interface* [name]
   |  |        ...
   +---n sav-event ⑤
   |  +--ro router-id?  yang:dotted-quad
   |        ...
```

SAVNET Configuration-General Information

Global configuration:

- Control for SAVNET (function enable/disable), SAV mode, sav-table capacity, SAV information source priorities.

Interface configuration:

- Control for SAVNET (function enable/disable), SAV mode, action of resetting SAV-statistics.

```
+--rw sav
   |  +--rw router-id?  yang:dotted-quad
   |  +--rw v4sav-entry-limits
   |  |  +--rw number?  uint32
   |  |  +--rw percent? uint8
   |  +--rw v6sav-entry-limits
   |  |  +--rw number?  uint32
   |  |  +--rw percent? uint8
   |  +--rw source-protocol-priorities
   |  |  +--rw source-protocol-priority* [type]
   |  |     +--rw type          identityref
   |  |     +--rw preference?   rule-preference
   |  +--rw sav-controls
   |  |  +--rw sav-enabled?   boolean
   |  |  +--rw sav-mode?      identityref
   |  |  +---x sav-interface-reset
   |  |  |  +---w input
   |  |  |     +---w reset-statistics?  Boolean
   augment /if:interfaces/if:interface:
   |  +--rw sav-control
   |  |  +--rw sav-enabled? boolean
   |  |  +--rw sav-mode?    identityref
   |  |  +---x sav-reset
   |  |  |  +---w input
   |  |  |     +---w reset-statistics?  boolean
```

SAVNET Configuration-Static SAV Rules

Static SAV rules configuration in the SAV table:

- SAV entries: “Source prefix” and “Incoming interface”, including IPv4 and IPv6 SAV rules.
- Capacity of SAV table and upper limitation of IPv4 or IPv6 SAV rules.

```
+--rw static-savs
|  +--rw v4sav:ipv4
|  |  +--rw v4sav:sav-entry-limits
|  |     +--rw v4sav:number?  uint32
|  |     +--rw v4sav:percent? uint8
|  |  +--rw v4sav:sav-rule* [source-prefix]
|  |     +--rw v4sav:source-prefix  inet:ipv4-prefix
|  |     +--rw v4sav:description?   string
|  |     +--rw v4sav:incoming-interfaces
|  |        +--rw v4sav:incoming-interface*
|  |        |       if:interface-ref
|  +--rw v6sav:ipv6
|  |  +--rw v6sav:sav-entry-limits
|  |     +--rw v6sav:number?  uint32
|  |     +--rw v6sav:percent? uint8
|  |  +--rw v6sav:sav-rule* [source-prefix]
|  |     +--rw v6sav:source-prefix  inet:ipv6-prefix
|  |     +--rw v6sav:description?   string
|  |     +--rw v6sav:incoming-interfaces
|  |        +--rw v6sav:incoming-interface*
|  |        |       if:interface-ref
```

SAVNET State-Sav Table

```
+--rw sav-tables
  +--rw sav-table* [name]
    +--ro name              string
    +--ro address-family?   identityref
    +--ro description?      string
    +--ro sav-rules
    |  +--ro sav-rule*
    |     +--ro rule-preference?       rule-preference
    |     +--ro incoming-interfaces
    |     |  +--ro incoming-interface*  if:interface-ref
    |     +--ro source-protocol        identityref
    |     +--ro active?                empty
    |     +--ro last-updated?          yang:date-and-time
    |     +--ro v4sav:source-prefix?    inet:ipv4-prefix
    |     +--ro v6sav:source-prefix?    inet:ipv6-prefix
    |     +--ro drop-packets?          unit64
    |     +--ro drop-bytes?            unit64
    |     +--ro sav-invalid-packets?    unit64
    |     +--ro sav-invalid-bytes?      unit64
    |     +--ro sav-valid-packets?      Unit64
    |     +--ro sav-valid-bytes?        unit64
    +---x active-sav-rule
          +---w input
          |  +---w v4sav:source-address?    inet:ipv4-address
          |  +---w v6sav:source-address?    inet:ipv6-address
          ……
```

The SAV table state:

- All SAV rules of IPv4 or IPv6 address families.
- Filtered SAV rules by the specified source prefix.
- SAV rule statistics and packets counts.

```
……

+--ro output
  +--ro sav-rule
    +--ro incoming-interfaces
    |  +--ro incoming-interface* if:interface-ref
    +--ro source-protocol        identityref
    +--ro active                 empty
    +--ro last-updated?          yang:date-and-time
    +--ro v4sav:source-prefix?   inet:ipv4-prefix
    +--ro v6sav:source-prefix?   inet:ipv6-prefix
    +--ro drop-packets?          unit64
    +--ro drop-bytes?            unit64
    +--ro sav-invalid-packets?   unit64
    +--ro sav-invalid-bytes?     unit64
    +--ro sav-valid-packets?     unit64
    +--ro sav-valid-bytes?       unit64
```

SAVNET State-Interface

The interface state:

- SAV statistics and packets counts. (valid/invalid /drop packets)

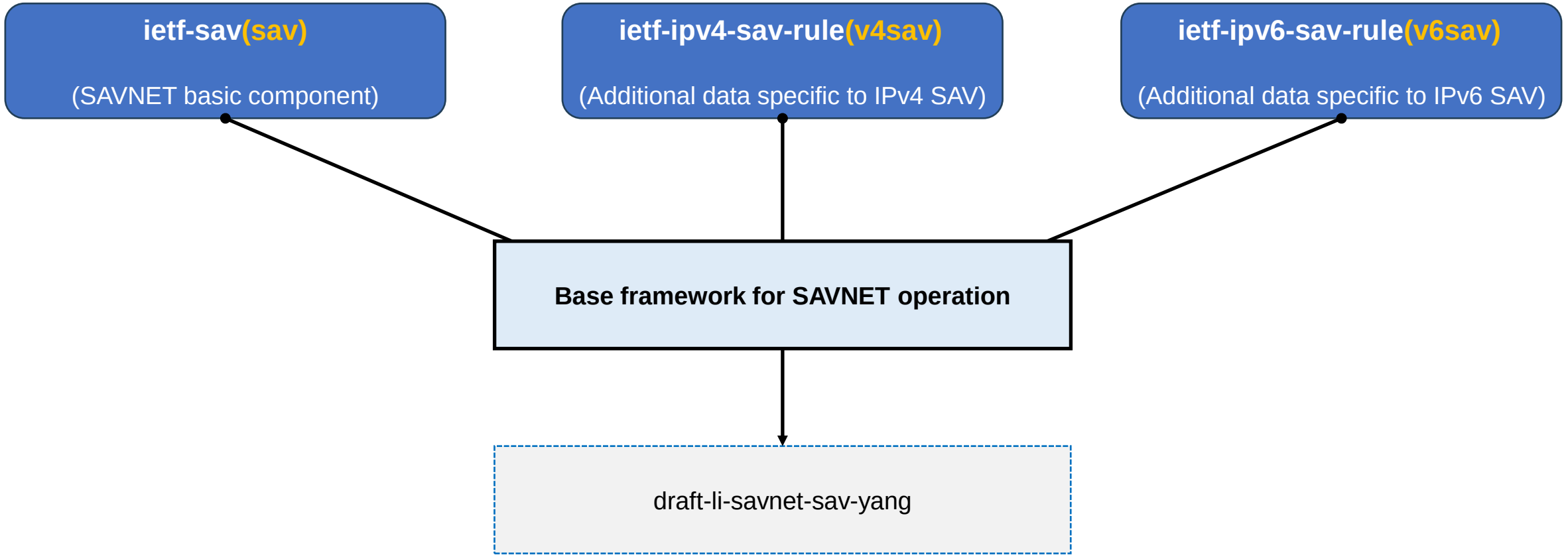
```
+--ro interfaces
|  +--ro interface* [name]
|     +--ro name          if:interface-ref
|     +--ro drop-packets? uint64
|     +--ro drop-bytes?   uint64
|     +--ro sav-invalid-packets? uint64
|     +--ro sav-invalid-bytes?   uint64
|     +--ro sav-valid-packets?   uint64
|     +--ro sav-valid-bytes?     uint64
```

SAVNET Event

Over-limitation of SAV table capacity should be notified.

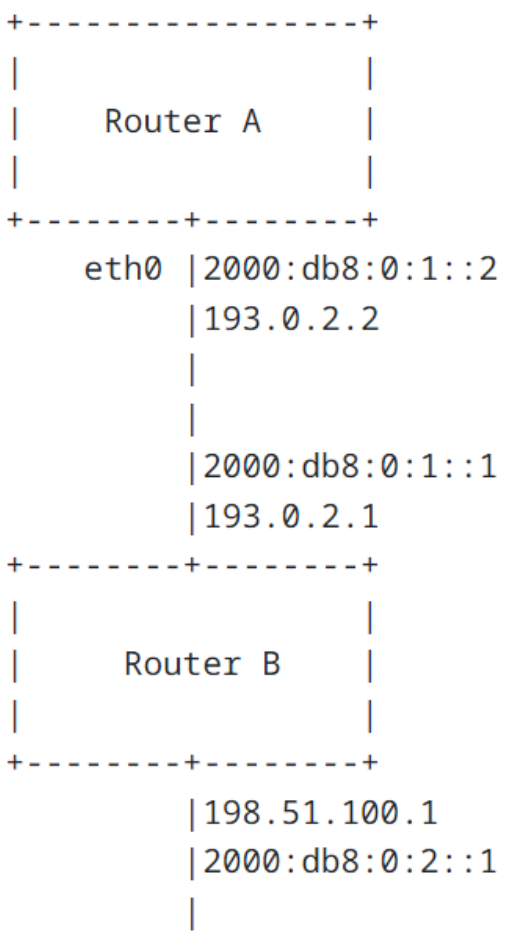
```
+---n sav-event
|  +--ro router-id?          yang:dotted-quad
|  +--ro address-family      identityref
|  +--ro sav-limit-reached?  boolean
```

SAVNET YANG Modules



- Structures of all nodes in tree diagram are defined in the above 3 modules (sav, v4sav, v6sav).
- Expected to be augmented by other SAV-related function modules accordingly.
- Division of the 3 modules also refers to the similar approach of RIB Yang in RFC8349.

Example-Static SAV Rules Configuration




```
"ietf-sav:sav": {
    "router-id": "193.0.2.2",
    "static-savs": {
      "ietf-ipv4-sav-rule:ipv4": {
        "sav-rule": [
          {
            "source-prefix": "198.51.100.0/24",
            "incoming-interfaces": {
              "incoming-interface": "eth0"
            }
          }
        ]
      },
      "ietf-ipv6-sav-rule:ipv6": {
        "sav-rule": [
          {
            "source-prefix": "2000:db8:0:2::/64",
            "incoming-interfaces": {
              "incoming-interface": "eth0"
            }
          }
        ]
      }
    },
```

Example: Statics SAV rule on Router-A is used to validating source prefixes of customer subnets attached to the edge Router-B

Example-Run Data


```
"sav-tables": {
      "sav-table": [
       {
         "name": "ipv4-master",
         "address-family":
           "ietf-sav:ipv4",
         "sav-rules": {
           "sav-rule": [
            {
              "ietf-ipv4-sav-rule:source-prefix":
                "198.51.100.0/24",
              "incoming-interfaces": {
                "incoming-interface": "eth0"
              },
              "rule-preference": 5,
              "source-protocol": "ietf-sav:static",
              "last-updated": "2023-5-20T17:11:27+02:00",
              "drop-packets": 0,
              "drop-bytes": 0,
              "sav-invalid-packets": 0,
              "sav-invalid-bytes": 0,
              "sav-valid-packets": 10,
              "sav-valid-bytes": 100
            }
           ]
         }
      }
```

```
{
        "name": "ipv6-master",
        "address-family":
          "ietf-sav:ipv6",
        "sav-rules": {
          "sav-rule": [
            {
              "ietf-ipv6-sav-rule:source-prefix":
                "2000:db8:0:2::/64",
              "incoming-interfaces": {
                "incoming-interface": "eth0"
              },
              "source-protocol": "ietf-routing:static",
              "route-preference": 5,
              "last-updated": "2023-5-20T17:11:27+02:00",
              "drop-packets": 0,
              "drop-bytes": 0,
              "sav-invalid-packets": 0,
              "sav-invalid-bytes": 0,
              "sav-valid-packets": 10,
              "sav-valid-bytes": 100
            }
          ]
        }
      }
```

Next Steps

- SAV rules from more types of control-plane-protocol. *[SAVNET architecture updates]*
- Request more review.
- Any comments welcomed.

Thanks!