

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: 3 December 2026

B. Wu
D. Dhody
Huawei Technologies
V.P. Beeram
Juniper Networks
T. Saad
Cisco Systems
S. Peng
ZTE Corporation
1 June 2026

YANG Data Models for Network Resource Partitions (NRPs)
draft-ietf-teas-nrp-yang-06

Abstract

RFC 9543 describes a framework for Network Slices in networks built from IETF technologies. In this framework, the network resource partition (NRP) is introduced as a collection of network resources allocated from the underlay network to carry a specific set of Network Slice Service traffic and meet specific Service Level Objective (SLO) and Service Level Expectation (SLE) characteristics.

This document defines two YANG data models for Network Resource Partitions (NRPs): a network-level model for policy configuration by a Network Slice Controller, and a device-level model for configuration of individual network elements. These models enable automated provisioning of NRPs in IP/MPLS and Segment Routing (SR) networks, supporting scalable realization of RFC 9543 Network Slice Services.

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 3 December 2026.

Copyright Notice

Copyright (c) 2026 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
2. Conventions and Definitions	3
3. NRP Data Models	5
3.1. Models Usage	5
3.2. NRPs Instantiation	6
3.2.1. Resource Reservation	8
3.2.2. NRP Selector Identifier (NRP Selector ID)	9
3.2.3. Per-Hop Behavior (PHB)	10
3.2.4. NRP Topology	11
3.3. NRPs Monitoring	12
3.4. NRPs Device Model Description	13
4. NRPs YANG Module	15
5. NRPs Device YANG Module	33
6. Scalability Considerations	37
7. Security Considerations	38
8. IANA Considerations	39
9. Acknowledgments	40
10. Contributor	40
11. References	41
11.1. Normative References	41
11.2. Informative References	43
Appendix A. Examples	46
A.1. Network-level NRP Policy with Hybrid Partition Example	47
A.2. Network-Level NRP Policy with Control-Plane Partition Example	49
A.3. Device-Level NRP Policy Example	50
Appendix B. NRPs YANG Module Tree	53
Appendix C. NRPs Device YANG Module Tree	57
Authors' Addresses	58

1. Introduction

[RFC9543] describes a framework for Network Slices in networks built from IETF technologies. As specified in Section 7.4 [RFC9543], an NRP is a collection of resources identified in the underlay network to support the RFC 9543 Network Slice Service to meet the slice Service Level Objectives (SLOs) and Service Level Expectations (SLEs) characteristics and network scalability. This document follows the Network Slice Service defined in [RFC9543], and all references to "Network Slice" refer to that same context.

Considering the NRPs realizations in IP/MPLS and Segment Routing (SR) networks, [I-D.ietf-teas-ns-ip-mpls] and [I-D.ietf-teas-nrp-scalability] describe NRP mechanisms of control plane, data plane, and management plane to provide specific forwarding treatment (scheduling, drop policy, resource usage) to the Slice Service packets associated with an NRP. Specifically, for instantiation of the device-specific and network wide NRPs, Section 3.5 of [I-D.ietf-teas-ns-ip-mpls] introduces the construct of NRP Policy.

Based on these descriptions of NRP, this document defines two YANG modules:

- * `ietf-nrp`: A network-level model used by a Network Slice Controller (NSC) (defined in Section 6.3 [RFC9543]) to manage NRP instances for Network Slice Service realizations. This is a network configuration model according to the YANG model classification of [RFC8309].
- * `ietf-nrp-device`: A device-level model used by a network controller to set NRP parameters on an individual device, including device-specific interface configurations. This is a device configuration model by the classification.

The NRPs models conform to the Network Management Datastore Architecture (NMDA) [RFC8342] and are designed to support both configuration and operational state retrieval.

2. Conventions and Definitions

The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP14, [RFC2119], [RFC8174] when, and only when, they appear in all capitals, as shown here.

The following terms are defined in [RFC6241], [RFC7950], and [RFC8345] and are used in this specification:

- * configuration data
- * state data
- * augment
- * data model
- * data node
- * network topology

The terminology for describing YANG data models is found in [RFC7950]. The tree diagram used in this document follows the notation defined in [RFC8340].

Additionally, this document uses the following terms:

Network Resource Partition (NRP): A collection of network resources allocated from the underlay network to support one or more RFC 9543 Network Slice Services, meeting specific SLO/SLE requirements. Defined in Section 7.1 of [RFC9543].

NRP Policy: A configuration construct that specifies the rules for creating and maintaining an NRP, including resource allocation, selector identification, and topology association.

NRP Instance: The operational realization of an NRP Policy on network devices, representing the actual allocated resources and configured forwarding behaviors.

NRP Selector ID: A data plane identifier used by network devices to classify packets belonging to a specific NRP and apply the corresponding forwarding treatment.

PHB Profile: A named set of Per-Hop Behavior parameters that define the forwarding treatment (e.g., scheduling, drop policy) for packets associated with an NRP. Profiles are locally defined by the network operator.

NRP-Enabled Device: A network element that supports NRP data plane identification and resource partitioning mechanisms.

Network Slice Controller (NSC): A functional component responsible for translating Network Slice Service requests into network configuration, including NRP management. Defined in Section 6.3 of [RFC9543].

3. NRP Data Models

3.1. Models Usage

As defined in Section 7.1 [RFC9543], the Network Slice Controller (NSC) can determine which specific connectivity constructs from one or more slices could be grouped together upon Slice Service requests. This could be based on a specific set of SLOs and SLEs, or on any administrative or operational policy. The NSC can further map these connectivity constructs onto an NRP. It also constructs and distributes the network wide consistent NRP model to the relevant controllers, and in turn the controllers distribute the NRP device model to the NRP-enabled devices in the underlay network. Figure 1 shows the interfaces to which the two models are applied.

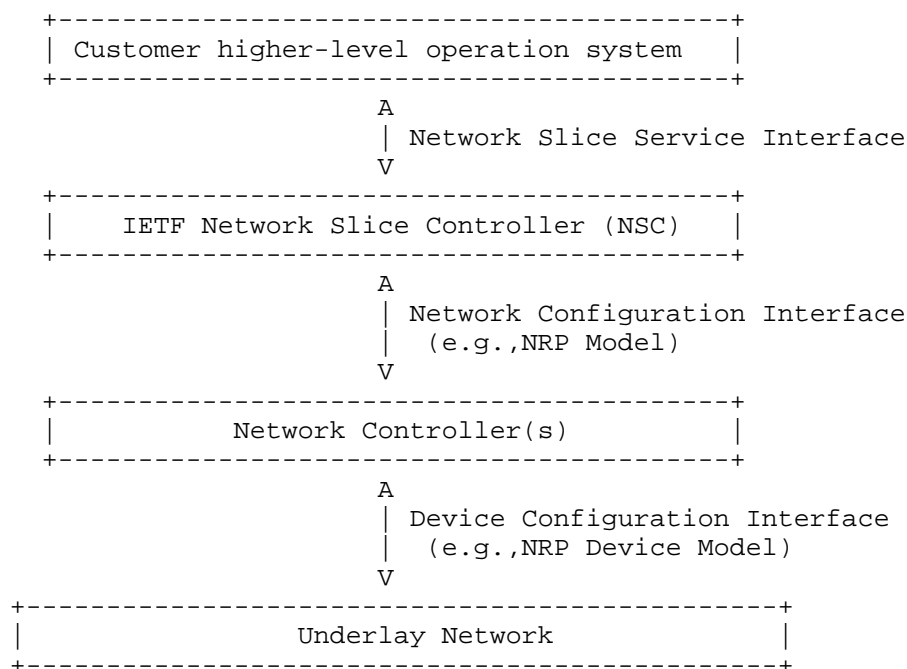


Figure 1

The general operations of NRPs are as follows:

- * NRPs instantiation: Depending on the Slice Service types, network resources status, and also the operator's policies, there can be at least two approaches. One method is to create an NRP instance before the network controller processes a Network Slice Service request. Another one is that the network controller may decide creating an NRP instance while implementing the Network Slice Service request.
- * NRPs modification: When the capacity of an existing NPR link is close to maximum capacity, the bandwidth of the link could be increased. And when an NRP links or nodes resources are insufficient, new NRP links and nodes could be added.
- * NRPs Deletion: If an NSC determines that no Slice Service is using an NRP, the NSC can delete the NRP instance. Another example is when it is necessary to merge NRPs.
- * NRPs Monitoring: The NSC can use the NRPs model to track and monitor NRPs resource status and usage.

3.2. NRPs Instantiation

An NRP Policy is a configuration construct that specifies the rules for creating and maintaining a Network Resource Partition. It defines how network resources are allocated, how traffic is identified, which topology is used, and what forwarding treatment is applied. These rules apply to both device-level and network-level configurations.

An NRP Policy includes the following key components:

1. Topology customization rules: Determine the set of nodes, links, and network functions that belong to the NRP.
2. Data plane rules: Specify the NRP selector ID, QoS profiles, and bandwidth reservations for traffic identification and forwarding treatment.
3. Control plane rules: Define resource reservation, resource sharing, reservation preferences policy, etc.

NRP Policies support three resource partition modes:

- a. Data Plane Partition: Physical network resources (e.g., bandwidth) are partitioned on network devices. Devices enable NRP data plane mechanisms to apply Per-Hop Behavior (PHB) to packets.

- b. Control Plane Partition: Resource reservation state for each NRP is maintained in NRP-aware TE devices or network controllers, without physical resource isolation.
- c. Hybrid Partition: Combines both data plane and control plane partitioning for full resource isolation and stateful management.

All modes require NRP topology specification. Modes (a) and (c) require NRP selector identification for packet classification. Modes (b) and (c) require distributed or centralized resource reservation management. When IGP routing is used with mode (c), NRP-aware IGP extensions may be employed for scalability.

For further implementation considerations, the concepts align with the NRP mechanisms described in [I-D.ietf-teas-ns-ip-mps] and [I-D.ietf-teas-nrp-scalability].

The high-level model structure of NRP policy as modeled in this document is shown in Figure 2:

```

module: ietf-nrp
  augment /nw:networks:
    +--rw nrp-policies
      +--rw qos-profiles
        |   +--rw phb-profile-identifier* [id]
        |   ...
      +--rw nrp-policy* [name]
        +--rw name                               string
        +--rw nrp-id?                             uint32
        +--rw mode?                               identityref
        +--rw resource-reservation
          |   ...
        +--rw selector-id
          |   ...
        +--rw phb-profile?                         leafref
        +--rw topology
          ...

```

Figure 2: NRP Policy subtree high-level structure

The "networks" container from the "ietf-network" module [RFC8345] provides a placeholder for an inventory of nodes in the network. This container is augmented to include a set of NRP policies.

The "qos-profiles" container provides a list of PHB policy profile entries. Each of these entries can be referenced by one or more NRPs. A "phb-profile-identifier" entry can have a reference to a standard PHB profile available on the device or the network controller.

The "nrp-policies" container includes a list of NRP policies. Each "nrp-policy" entry is identified by a name and holds the set of attributes needed to instantiate an NRP.

The description of the "nrp-policies" data nodes are as follows, and the other key elements of each nrp-policy entry are discussed in the following sub-sections.

- * "nrp-id": Is an identifier that is used to uniquely identify an NRP instance within an NSC network scope, which is created by the enforcement of the "nrp-policy".
- * "mode": Refers to control plane partition, data plane partition, or hybrid partition.

3.2.1. Resource Reservation

The "resource-reservation" container specifies bidirectional bandwidth allocation for an NRP, expressed as either an absolute value or a percentage of link capacity. While currently limited to bandwidth, the container is designed to accommodate future extensions such as resource sharing or protection mechanisms described in [I-D.ietf-teas-ns-ip-mpls].

The bandwidth reservation specified at the policy level serves as a default for all links in the NRP topology. This can be overridden by link-specific configuration in the "nrp-topology" container Section 3.2.4.

```

+--rw resource-reservation
  +--rw (max-bw-type)?
    +--:(bw-value)
      |   +--rw maximum-bandwidth?          uint64
    +--:(bw-percentage)
      +--rw maximum-bandwidth-percent?    rt-types:percentage

```

Figure 3: NRP Resource Reservation YANG subtree structure

3.2.2. NRP Selector Identifier (NRP Selector ID)

The NRP selector ID defines data plane encapsulation types and values used to identify NRP-specific network resources. The configuration can be overridden by link-specific "selector-id" configuration in "nrp-topology" (Section 3.2.4).

Implementations MUST support at least one selector type in the NRP model of data plane partition or hybrid partition. Multiple selector types MAY be configured to support heterogeneous network environments. Selector ID values MUST be unique within the network scope where the NRP operates.

When the "selector" feature is supported, the selector-id can be configured using one or more of the following types:

IPv4-derived: Uses destination IPv4 prefixes to identify NRP traffic. Suitable for networks where NRPs are aligned with IP addressing.

IPv6: Provides three sub-options:

- Dedicated: Uses a value in the IPv6 Hop-by-Hop extension header as specified in [I-D.ietf-6man-enhanced-vpn-vtn-id]

- SRv6-derived: Uses SRv6 SIDs where the SID structure encodes the NRP identifier, per [I-D.ietf-spring-sr-for-enhanced-vpn].

- IPv6-destination-derived: Uses destination IPv6 prefixes, similar to IPv4-derived.

MPLS: Supports two encoding mechanisms:

- In-Stack MPLS Network Action (MNA): Embeds the selector in the MPLS label stack using the format defined in [I-D.ietf-mpls-mna-nrp-selector]. Supports 13-bit (NRPS13) or 20-bit (NRPS20/ENRPS20) encodings.

- Post-Stack (PSD): Uses a dedicated post-stack label per [I-D.ietf-mpls-mna-psd-nrp-selector].

ACL-based: References standard ACLs to match NRP traffic. Provides flexibility for complex classification rules but may have higher processing overhead.

```

+--rw selector-id
|   +--rw ipv4-derived {ipv4-selector}?
|   |   +--rw destination-prefix*   inet:ipv4-prefix
|   +--rw ipv6
|   |   {ipv6-hbh-selector or srv6-selector or
ipv6-destination-selector}?
|   |   +--rw (selector-type)?
|   |   |   +--:(dedicated) {ipv6-hbh-selector}?
|   |   |   |   +--rw ipv6-hbh-eh?           uint32
|   |   |   +--:(srv6-derived) {srv6-selector}?
|   |   |   |   +--rw srv6-sid*             srv6-types:srv6-sid
|   |   |   +--:(ipv6-destination-derived)
|   |   |   |   {ipv6-destination-selector}?
|   |   |   |   +--rw destination-prefix*   inet:ipv6-prefix
|   +--rw mpls
|   |   {mpls-mna-selector or mpls-post-stack-selector}?
|   |   +--rw (mpls-encapsulation)?
|   |   |   +--:(mna-in-stack) {mpls-mna-selector}?
|   |   |   |   +--rw in-stack-identifier?   uint32
|   |   |   |   +--rw identifier-format?     enumeration
|   |   |   +--:(post-stack) {mpls-post-stack-selector}?
|   |   |   |   +--rw post-stack-identifier? uint32
|   +--rw acl-ref*           acl-ref {acl-selector}?

```

Figure 4: NRP Selector ID YANG subtree structure

3.2.3. Per-Hop Behavior (PHB)

Per-Hop Behavior (PHB) defines the forwarding treatment for packets belonging to an NRP, including bandwidth control, congestion control, and scheduling priority (e.g., Section 3.4 [RFC3644]). The PHB profile and NRP selector are paired mechanisms: the selector classifies packets, and the referenced PHB profile applies the corresponding forwarding behavior. Profile contents are implementation-specific and may include standard IETF PHBs (e.g., "Assured Forwarding (AF)", "Expedited Forwarding (EF)") or customized local policies (e.g., "High", "Low", "Standard").

The phb-profile configured at the NRP policy level serves as the default for the entire NRP topology. This may be overridden by link-specific PHB configuration in the topology container Section 3.2.4.

```

+--rw phb-profile?           leafref

```

Figure 5: PHB YANG subtree structure

3.2.4. NRP Topology

"nrp-topology" defines a customized NRP topology used for an NRP.

When an NRP supports IGP routing, the topology of the NRP must be congruent with an IGP instance. The topology used for IGP route computation and forwarding can be derived using Multi-Topology Routing (MTR) [RFC4915], [RFC5120], and [I-D.ietf-lsr-isis-sr-vtn-mt] or Flex-algo [RFC9350].

Figure 6 shows an example of NRP-1 enabling "igp-congruent", which indicates that this NRP instance uses the same IGP topology with the specified "multi-topology-id" or "algo-id". NRP-1 has different link resource attributes from those of the IGP, but shares the same nodes and termination points (TPs) of the IGP topology.

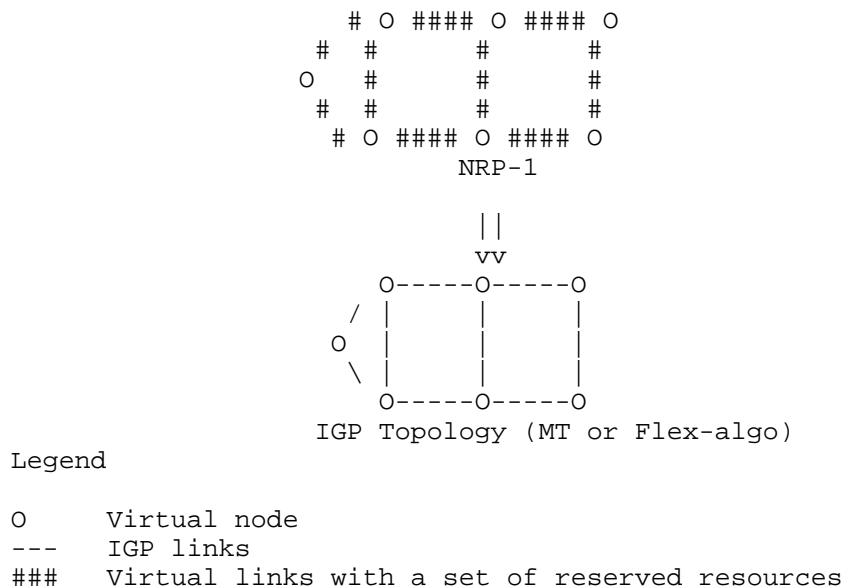


Figure 6: IGP Congruency Example

The "selection" container consists of a list of select subset of links of an underlay topology or a pre-built topology.

The "filter" container consists of a list of filters where each entry references a topology filter [I-D.ietf-teas-yang-topology-filter]. The topological elements that satisfy the membership criteria may override the default "resource-reservation" and "selector-id" specific nodes.

```

+--rw topology
  +--rw igp-congruent!
  |   +--rw multi-topology-id?   uint32
  |   +--rw algo-id?             uint32
  |   +--rw sharing?             boolean
  +--rw (topology-type)?
  |   +--:(selection)
  |   |   +--rw select
  |   |   |   +--rw topology-group* [group-id]
  |   |   |   |   +--rw group-id             string
  |   |   |   |   +--rw network-ref?
  |   |   |   |   |   -> /nw:networks/network/network-id
  |   |   |   +--rw link* [link-ref]
  |   |   |   |   ...
  |   |   |   +--rw resource-reservation
  |   |   |   |   ...
  |   |   |   +--rw link-partition-type?     identityref
  |   |   |   +--rw phb-profile?             leafref
  |   +--:(filter)
  |   |   +--rw filters
  |   |   |   +--rw filter* [filter-ref]
  |   |   |   |   +--rw filter-ref
  |   |   |   |   |   topo-filter-ref
  |   |   |   +--rw resource-reservation
  |   |   |   |   ...
  |   |   |   +--rw selector-id
  |   |   |   |   ...
  |   |   |   +--rw phb-profile?             leafref

```

Figure 7: NRP Topology YANG subtree structure

3.3. NRPs Monitoring

The NRP model can be used to monitor the operational status and resource usage of NRPs.

```

augment /nw:networks/nw:network/nw:network-types:
  +--rw nrp!
augment /nw:networks/nw:network/nw:node:
  +--ro (nrp-attachment-type)?
    +--:(single-nrp)
      | +--ro nrp-node-attributes
      | | +--ro selector-id
      | | | +--ro srv6?          srv6-types:srv6-sid
      | | | +--ro mpls-nrp-selector?  uint32
      | +--:(multi-nrp)
      | | +--ro node-nrp-list
      | | | +--ro nrp* [nrp-id]
      | | | | +--ro nrp-id          uint32
      | | | | +--ro nrp-node-attributes
      | | | | ...
    +--:(single-nrp)
      | +--ro nrp-link-attributes
      | | +--ro link-partition-type?  identityref
      | | +--ro bandwidth-value?      uint64
      | | +--ro selector-id
      | | | +--ro srv6?          srv6-types:srv6-sid
      | | | +--ro mpls-nrp-selector?  uint32
      | | +--ro statistics
      | | | +--ro status
      | | | | ...
      | | | +--ro one-way-available-bandwidth?  uint64
      | | | +--ro one-way-utilized-bandwidth?   uint64
      | | | +--ro one-way-min-delay?            uint32
      | | | +--ro one-way-max-delay?            uint32
      | | | +--ro one-way-delay-variation?      uint32
      | | | +--ro one-way-packet-loss?          decimal64
    +--:(multi-nrp)
      | +--ro link-nrp-list
      | | +--ro nrp* [nrp-id]
      | | | +--ro nrp-id          uint32
      | | | +--ro nrp-link-attributes
      | | | ...

```

Figure 8: NRPs Monitoring YANG subtree structure

3.4. NRPs Device Model Description

The device-specific NRPs model is defined in module "ietf-nrp-device" as shown in Section 5.

The NRP device YANG data model is only applicable to device configuration and includes attributes such as QoS policies, resource reservations, and NRP selector IDs. Specifically, it adds interface-specific attributes for cases where the QoS policies, NRP resources, and NRP selector IDs of an interface differ from the global NRP attributes of the device.

Figure 9 shows the tree diagram of the device NRPs YANG model defined in modules "ietf-nrp-device.yang".

```

module: ietf-nrp-device
  +--rw nrp-policies
    +--rw qos-profiles
      |   +--rw phb-profile-identifier* [id]
      |   +--rw id string
    +--rw nrp-policy* [name]
      +--rw name string
      +--rw nrp-id? uint32
      +--rw resource-reservation
        |   +--rw (max-bw-type)?
        |   ...
      +--rw selector-id
        |   +--rw ipv4-derived {ipv4-selector}?
        |   |   ...
        |   +--rw ipv6
        |   |   {ipv6-hbh-selector or srv6-selector or
        |   |   ipv6-destination-selector}?
        |   |   ...
        |   +--rw mpls
        |   |   {mpls-mna-selector or mpls-post-stack-selector}?
        |   |   ...
        |   +--rw acl-ref* acl-ref {acl-selector}?
      +--rw phb-profile? leafref
      +--rw igp-congruent!
        |   +--rw multi-topology-id? uint32
        |   +--rw algo-id? uint32
        |   +--rw sharing? boolean
      +--rw interfaces
        +--rw interface* [interface]
        ...
  
```

Figure 9: NRPs Device YANG subtree high-level structure

4. NRPs YANG Module

The "ietf-nrp" module uses types defined in [RFC8345], [RFC8294], [RFC8776], [RFC9911], [RFC8519], [RFC9181], [I-D.ietf-spring-srv6-yang-base], and [I-D.ietf-teas-yang-topology-filter].

```
<CODE BEGINS> file "ietf-nrp@2026-06-01.yang"
module ietf-nrp {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-nrp";
  prefix nrp;

  import ietf-network {
    prefix nw;
    reference
      "RFC 8345: A YANG Data Model for Network Topologies,
       Section 6.1";
  }
  import ietf-network-topology {
    prefix nt;
    reference
      "RFC 8345: A YANG Data Model for Network Topologies,
       Section 6.2";
  }
  import ietf-routing-types {
    prefix rt-types;
    reference
      "RFC 8294: Common YANG Data Types for the Routing Area";
  }
  import ietf-vpn-common {
    prefix vpn-common;
    reference
      "RFC 9181: A Common YANG Data Model for Layer 2 and Layer 3
       VPNs";
  }
  import ietf-te-packet-types {
    prefix te-packet-types;
    reference
      "RFC 8776: Traffic Engineering Common YANG Types";
  }
  import ietf-inet-types {
    prefix inet;
    reference
      "RFC 9911: Common YANG Data Types";
  }
  import ietf-access-control-list {
    prefix acl;
```

```
reference
  "RFC 8519: YANG Data Model for Network Access Control Lists
    (ACLs)";
}
import ietf-srv6-types {
  prefix srv6-types;
  reference
    "draft-ietf-spring-srv6-yang-base: YANG Data Model for SRv6 Base";
}
import ietf-topology-filter {
  prefix topo-filt;
  reference
    "draft-ietf-teas-yang-topology-filter: YANG Data Model
      for Topology Filter";
}

organization
  "IETF Traffic Engineering Architecture and Signaling (TEAS)
    Working Group";
contact
  "WG Web:    <https://datatracker.ietf.org/wg/teas/>
  WG List:    <mailto:teas@ietf.org>

  Editor:     Bo Wu
              <mailto:lane.wubo@huawei.com>

  Editor:     Dhruv Dhody
              <mailto:dhruv.ietf@gmail.com>

  Editor:     Vishnu Pavan Beeram
              <mailto:vbeeram@juniper.net>

  Editor:     Tarek Saad
              <mailto:tsaad.net@gmail.com>

  Editor:     Shaofu Peng
              <mailto:peng.shaofu@zte.com.cn>";
description
  "This YANG module defines a data model for
    Network Resource Partitions (NRPs) management.

    Copyright (c) 2026 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
(<https://www.rfc-editor.org/info/rfcXXXX>); see the RFC itself
for full legal notices.";

```
revision 2026-06-01 {
  description
    "Initial revision.";
  reference
    "RFC XXXX: YANG Data Models for Network Resource
      Partitions (NRPs)";
}

/* Features - NRP Selector Mechanisms */

feature ipv4-selector {
  description
    "Indicates support for IPv4-derived NRP selection
      (destination prefix based).";
}

feature ipv6-hbh-selector {
  description
    "Indicates support for dedicated IPv6 Hop?by?Hop
      extension header selector ID.";
}

feature srv6-selector {
  description
    "Indicates support for SRv6 SID?based NRP selection.";
}

feature ipv6-destination-selector {
  description
    "Indicates support for IPv6 destination prefix?based
      NRP selection.";
}

feature mpls-mna-selector {
  description
    "Indicates support for in?stack MPLS MNA?based
      NRP selection (NRPS13, NRPS20, ENRPS20).";
}

feature mpls-post-stack-selector {
  description
```

```
        "Indicates support for post?stack MPLS PSD?based
        NRP selection.";
    }

    feature acl-selector {
        description
            "Indicates support for ACL?based NRP selection.";
    }

    /*
    * I D E N T I T I E S
    */

    identity nrp-partition-mode {
        description
            "Base identity for NRP partition type.";
    }

    identity control-plane-partition {
        base nrp-partition-mode;
        description
            "NRP control plane partition.";
    }

    identity data-plane-partition {
        base nrp-partition-mode;
        description
            "NRP data plane partition.";
    }

    identity hybrid-plane-partition {
        base nrp-partition-mode;
        description
            "Both control and data planes partitions of NRP.";
    }

    identity nrp-link-partition-type {
        description
            "Base identity for NRP interface partition type.";
    }

    identity virtual-interface {
        base nrp-link-partition-type;
        description
            "Identity for NRP virtual interface or sub-interface partition,
            e.g., FlexE.";
    }
```

```
identity queue-partition {
  base nrp-link-partition-type;
  description
    "Identity for NRP queue partition type.";
}

/*
 * T Y P E D E F S
 */

typedef acl-ref {
  type leafref {
    path "/acl:acls/acl:acl/acl:name";
  }
  description
    "Used to reference an ACL.";
}

typedef topo-filter-ref {
  type leafref {
    path "/nw:networks/topo-filt:topology-filters/"
      + "topo-filt:topology-filter/topo-filt:name";
  }
  description
    "This type is used to reference a Topology Filter.";
  reference
    "draft-ietf-teas-yang-topology-filter: YANG Data Model
    for Topology Filter";
}

/*
 * Grouping - NRP Resource Reservation
 */

grouping resource-reservation {
  description
    "Grouping for NRP resource reservation.";
  container resource-reservation {
    description
      "Container for NRP resource reservation.";
    choice max-bw-type {
      description
        "Choice of maximum bandwidth specification.";
      case bw-value {
        leaf maximum-bandwidth {
          type uint64;
          units "bits/second";
          description
```

```

        "The maximum bandwidth allocated to an NRP
        - specified as absolute value.";
    }
}
case bw-percentage {
    leaf maximum-bandwidth-percent {
        type rt-types:percentage;
        description
            "The maximum bandwidth allocated to an NRP
            - specified as percentage of link
            capacity.";
    }
}
}
}
}
}

/*
 * Grouping - NRP Selector Identifier Configuration
 */

grouping selector-id {
    description
        "Grouping for NRP selector identifier (NRP Selector ID)
        configuration.";
    container selector-id {
        description
            "Container for NRP selector ID. It can contain one or more
            selector mechanisms (IPv4 prefixes, IPv6 Hop-by-Hop options,
            SRv6 SIDs, MPLS labels, or ACLs) depending on the features
            supported by the implementation.";
        container ipv4-derived {
            if-feature "ipv4-selector";
            description
                "Container for IPv4 NRP selector ID.";
            leaf-list destination-prefix {
                type inet:ipv4-prefix;
                description
                    "Any prefix from the specified set of IPv4
                    destination prefixes can be the selector ID.";
            }
        }
        container ipv6 {
            if-feature
                "ipv6-hbh-selector or srv6-selector or
                ipv6-destination-selector";
            description
                "Container for IPv6 NRP selector ID.";
        }
    }
}

```

```
choice selector-type {
  description
    "Choices for IPv6 selector ID type.";
  case dedicated {
    if-feature "ipv6-hbh-selector";
    leaf ipv6-hbh-eh {
      type uint32;
      description
        "The selector ID carried in Hop-by-Hop option of
        IPv6 extension header.";
      reference
        "draft-ietf-6man-enhanced-vpn-vtn-id: Carrying
        Network Resource (NRP) related Information in IPv6
        Extension Header";
    }
  }
  case srv6-derived {
    if-feature "srv6-selector";
    leaf-list srv6-sid {
      type srv6-types:srv6-sid;
      description
        "Any SID from the specified set of SRv6 SID can
        be the NRP selector ID.";
      reference
        "draft-ietf-spring-sr-for-enhanced-vpn: Segment
        Routing based Network Resource Partition (NRP) for
        Enhanced VPN";
    }
  }
  case ipv6-destination-derived {
    if-feature "ipv6-destination-selector";
    leaf-list destination-prefix {
      type inet:ipv6-prefix;
      description
        "Any prefix from the specified set of IPv6
        destination prefixes can be the NRP selector ID.";
    }
  }
}

container mpls {
  if-feature "mpls-mna-selector or mpls-post-stack-selector";
  description
    "Container for MPLS NRP selector ID. Supports both
    In-Stack (MNA) and Post-Stack (PSD) encoding mechanisms.";
  choice mpls-encapsulation {
    description
      "Selects the MPLS NRP Selector encoding mechanism.";
```

```
case mna-in-stack {
  if-feature "mpls-mna-selector";
  description
    "In-Stack NRP Selector action.
     Refers to draft-ietf-mpls-mna-nrp-selector.";
  leaf in-stack-identifier {
    type uint32;
    must
      "(../identifier-format='nrps13' and . <= 8191) or
       (../identifier-format='nrps20' and . <= 1048575) or
       (../identifier-format='enrps20' and . <= 1048575)" {
        error-message
          "NRP Selector value exceeds format range";
      }
    description
      "The NRP Selector value. The valid range depends on the
       encoding format (13-bit or 20-bit) selected.";
  }
  leaf identifier-format {
    type enumeration {
      enum nrps13 {
        description
          "13-bit NRP Selector (NRPS13).
           Value range: 0-8191.";
      }
      enum nrps20 {
        description
          "20-bit NRP Selector (NRPS20).
           Value range: 0-1048575.";
      }
      enum enrps20 {
        description
          "20-bit Extended NRP Selector (ENRPS20) compatible
           with Entropy Label. Value range: 0-1048575.";
      }
    }
    description
      "Specifies the encoding format for the In-Stack
       NRP Selector.";
  }
}
}
case post-stack {
  if-feature "mpls-post-stack-selector";
  description
    "Post-Stack NRP Selector (PS-NRP).
     Refers to draft-ietf-mpls-mna-psd-nrp-selector.";
  leaf post-stack-identifier {
    type uint32 {
```

```

        range "0..4294967295"; // 2^32 - 1
    }
    description
        "The 32-bit Post-Stack NRP Selector.";
    }
}
}
leaf-list acl-ref {
    if-feature "acl-selector";
    type acl-ref;
    description
        "NRP selection is done based on the specified list of
        ACLs.";
    reference
        "RFC 8519: YANG Data Model for Network Access Control Lists
        (ACLs)";
}
}
}

/*
 * Grouping - NRP QoS Per-Hop Behavior (PHB) profiles
 */

grouping qos-profiles {
    description
        "Grouping for NRP QoS profiles.";
    container qos-profiles {
        description
            "Container for profiles.";
        list phb-profile-identifier {
            key "id";
            description
                "List of PHB profiles.";
            leaf id {
                type string {
                    length "1..max";
                }
            }
            description
                "Uniquely identifies the PHB.
                The profile only has significance within the service
                provider's administrative domain";
        }
    }
}
}

```

```
/*
 * Grouping - NRP QoS Per-Hop Behavior (PHB) profile
 */

grouping phb-profile-ref {
  description
    "Grouping for NRP QoS Per-Hop Behavior (PHB) profile.";
  leaf phb-profile {
    type leafref {
      path "/nw:networks/nrp:nrp-policies/"
        + "nrp:qos-profiles/nrp:phb-profile-identifier/nrp:id";
    }
  }
  description
    "PHB profile identifier, specifying the forwarding treatment
    of packets belonging to a specific NRP selector ID, such as
    bandwidth control, congestion control
    (e.g., Section 3.4 of [RFC3644]).
    The PHB may be standard PHB, such as Assured Forwarding (AF),
    Expedited Forwarding (EF), or a customized local policy,
    such as 'High', 'Low', 'Standard'.";
}
}

/*
 * Grouping - NRP IGP congruent
 */

grouping igp-congruent {
  description
    "Grouping for NRP IGP congruent attributes.";
  container igp-congruent {
    presence "Indicates NRP IGP congruency.";
    description
      "The presence of the container node describes NRP IGP
      congruent, which indicates that the NRP instance uses the
      same IGP topology with the specified 'multi-topology-id'
      and/or 'algo-id'. That is, the nodes and termination point
      of the NRP topology and the IGP topology are the same,
      while the link attributes of the NRP are different from
      those of the IGP.
      If not explicitly configured, both
      (multi-topology-id, algo-id) pair default to 0 (standard
      topology with standard SPF algorithm), corresponding to
      the base IGP instance.";
    leaf multi-topology-id {
      type uint32;
      default "0";
      description

```



```
    "Indicates the MT-id of the NRP IGP instance.
    MT-id 0 is reserved for the standard topology";
  reference
    "RFC 5120: M-ISIS: Multi Topology (MT) Routing in
    Intermediate System to Intermediate Systems (IS-ISs)
    RFC 4915: Multi-Topology (MT) Routing in OSPF";
}
leaf algo-id {
  type uint32;
  default "0";
  description
    "Indicates the algo-id of the NRP IGP instance.
    algo-id 0 is reserved for the standard SPF algorithm";
  reference
    "RFC 9350: IGP Flexible Algorithm";
}
leaf sharing {
  type boolean;
  description
    "'true' if the NRP IGP instance can be shared with
    other NRPs;
    'false' if the NRP IGP instance is dedicated
    to this NRP.";
}
}
}

/*
 * Grouping - NRP Topology Filter
 */

grouping topology-filter {
  description
    "Grouping for NRP filter topology.";
  container filters {
    description
      "Container for filters.";
    list filter {
      key "filter-ref";
      description
        "List of filters.";
      leaf filter-ref {
        type topo-filter-ref;
        description
          "Reference to a specific topology filter from the
          list of global topology filters.";
      }
    }
  }
  uses resource-reservation;
```

```

        uses selector-id;
        uses phb-profile-ref;
    }
}

/*
 * Grouping - NRP Select Topology
 */

grouping select-topology {
    description
        "NRP topology specified by selection.";
    container select {
        description
            "The container of NRP select topology.";
        list topology-group {
            key "group-id";
            description
                "List of groups for NRP topology elements (node or links)
                 that share common attributes.";
            leaf group-id {
                type string {
                    length "1..max";
                }
                description
                    "The NRP topology group identifier.";
            }
        }
        /*The base topology reference. All links in
        the list below must belong to this specific
        network.*/
        uses nw:network-ref;
        list link {
            key "link-ref";
            description
                "A list of links with common attributes that are
                 part of the referenced network";
            leaf link-ref {
                type leafref {
                    path "/nw:networks/nw:network"
                      + "[nw:network-id=current()/../../nrp:network-ref]"
                      + "/nt:link/nt:link-id";
                }
                description
                    "A reference to a link in the base topology specified
                     by the 'network-ref' sibling node.";
            }
        }
    }
}

```

```
    uses resource-reservation;
    leaf link-partition-type {
        type identityref {
            base nrp-link-partition-type;
        }
        description
            "Indicates the resource reservation type of an NRP link.";
    }
    uses phb-profile-ref;
}
}

/*
 * Grouping - NRP Policy
 */

grouping nrp-pol {
    description
        "Grouping for NRP policies.";
    container nrp-policies {
        description
            "Container for nrp policies.";
        uses qos-profiles;
        list nrp-policy {
            key "name";
            unique "nrp-id";
            description
                "List of NRP policies.";
            leaf name {
                type string {
                    length "1..max";
                }
                description
                    "A string that uniquely identifies the NRP policy.";
            }
            leaf nrp-id {
                type uint32;
                mandatory true;
                description
                    "A 32-bit ID that uniquely identifies the NRP
                     created by the enforcement of this NRP policy.";
            }
            leaf mode {
                type identityref {
                    base nrp-partition-mode;
                }
                mandatory true;
            }
        }
    }
}
```

```

        description
            "Indicates the resource partition mode of the NRP, such
            as control plane partition, data plane partition,
            or hybrid partition.";
    }
    uses resource-reservation;
    uses selector-id;
    uses phb-profile-ref;
    container topology {
        description
            "Container for NRP topology.";
        uses igp-congruent;
        choice topology-type {
            description
                "Choice of NRP topology type.";
            case selection {
                uses select-topology;
            }
            case filter {
                uses topology-filter;
            }
        }
    }
}

/*
 * Grouping - NRP Selector ID State
 */

grouping selector-id-state {
    description
        "The grouping of NRP selector ID state.";
    container selector-id {
        config false;
        description
            "The container of NRP selector ID.";
        leaf srv6 {
            type srv6-types:srv6-sid;
            description
                "SRv6 SID value.";
        }
        leaf mpls-nrp-selector {
            type uint32;
            description
                "MPLS NRP selector value.";
        }
    }
}

```

```
    }
  }

  // nrp-link-statistics

  grouping statistics-per-link {
    description
      "Statistics attributes per NRP link.";
    container statistics {
      config false;
      description
        "Statistics for NRP link.";
      uses vpn-common:service-status;
      uses nrp-bandwidth-metrics;
      uses te-packet-types:one-way-performance-metrics-packet;
    }
  }

  /*
   * Grouping - NRP node attributes
   */

  grouping node-nrp-attributes {
    description
      "NRP node scope attributes.";
    container nrp-node-attributes {
      config false;
      description
        "Containing NRP attributes.";
      uses selector-id-state;
    }
  }

  /*
   * Grouping - NRP Link Attributes
   */

  grouping link-nrp-attributes {
    description
      "NRP link scope attributes.";
    container nrp-link-attributes {
      config false;
      description
        "Contains NRP link attributes.";
      leaf link-partition-type {
        type identityref {
          base nrp-link-partition-type;
        }
      }
    }
  }
}
```

```
        description
            "Indicates the resource partition type of an NRP link.";
    }
    leaf bandwidth-value {
        type uint64;
        units "bits/second";
        description
            "Bandwidth allocation for the NRP as absolute value.";
    }
    uses selector-id-state;
    uses statistics-per-link;
}

/*
 * Grouping - NRP Bandwidth Metrics
 */

grouping nrp-bandwidth-metrics {
    description
        "Grouping for NRP bandwidth metrics.";
    leaf one-way-available-bandwidth {
        type uint64;
        units "bits/second";
        description
            "Available bandwidth that is defined to be NRP link
            bandwidth minus bandwidth utilization.";
    }
    leaf one-way-utilized-bandwidth {
        type uint64;
        units "bits/second";
        description
            "Bandwidth utilization that represents the actual
            utilization of the link (i.e., as measured in the router).";
    }
}

/*
 * Grouping - NRPs Node Attributes
 */

grouping node-nrp-list-attributes {
    description
        "Grouping for NRPs nodes attributes.";
    container node-nrp-list {
        config false;
        description
            "Containing NRPs attributes.";
```

```
list nrp {
  key "nrp-id";
  description
    "List of NRPs.";
  leaf nrp-id {
    type uint32;
    description
      "NRP identifier";
  }
  uses node-nrp-attributes;
}
}

/*
 * Grouping - NRPs Link Attributes
 */

grouping link-nrp-list-attributes {
  description
    "Grouping for NRPs link attributes.";
  container link-nrp-list {
    config false;
    description
      "Contains NRPs link attributes.";
    list nrp {
      key "nrp-id";
      description
        "List of NRPs.";
      leaf nrp-id {
        type uint32;
        description
          "NRP identifier";
      }
      uses link-nrp-attributes;
    }
  }
}

// nrp-network-type

grouping nrp-network-type {
  description
    "Identifies the network type to be NRP.";
  container nrp {
    presence "Indicates NRP network topology.";
    description
      "The presence of the container node indicates NRP network.";
  }
}
```

```
    }  
  }  
  
  /*  
  * Augment - Network Resource Partition Policies.  
  */  
  
  augment "/nw:networks" {  
    description  
      "Augments networks with NRP policies.";  
    uses nrp-pol;  
  }  
  
  /*  
  * Augment - NRP type.  
  */  
  
  augment "/nw:networks/nw:network/nw:network-types" {  
    description  
      "Indicates the network type of NRP";  
    uses nrp-network-type;  
  }  
  
  /*  
  * Augment - NRP node operational status.  
  */  
  
  augment "/nw:networks/nw:network/nw:node" {  
    description  
      "Augments node with NRP state attributes.";  
    choice nrp-attachment-type {  
      config false;  
      description  
        "Determines how NRP information is attached to node:  
        single-nrp for NRP-specific topology, multi-nrp for  
        native topology with multiple NRPs.";  
      case single-nrp {  
        when '../nw:network-types/nrp:nrp' {  
          description  
            "Condition: only applies to NRP network topologies.";  
        }  
        uses node-nrp-attributes;  
      }  
      case multi-nrp {  
        description  
          "Native topology with NRPs node operational status.";  
        uses node-nrp-list-attributes;  
      }  
    }  
  }  
}
```



```

    }
  }

  /*
   * Augment - NRP link operational status.
   */

  augment "/nw:networks/nw:network/nt:link" {
    description
      "Augments link with NRP state attributes.";
    choice nrp-attachment-type {
      config false;
      description
        "Determines how NRP information is attached to link:
         single-nrp for NRP-specific topology, multi-nrp for
         native topology with multiple NRPs.";
      case single-nrp {
        when '../nw:network-types/nrp:nrp' {
          description
            "Condition: only applies to NRP network topology.";
        }
        uses link-nrp-attributes;
      }
      case multi-nrp {
        /*Native topology with NRPs link operational status.*/
        description
          "Native topology with NRPs link operational status.";
        uses link-nrp-list-attributes;
      }
    }
  }
}
<CODE ENDS>

```

5. NRPs Device YANG Module

The NRP YANG module for devices ("ietf-nrp-device") reuses the qos policy, "resource-reservation", "selector-id" grouping defined in Section 4, and adds interface-specific NRP attributes.

The device NRPs YANG module also imports the following module(s): ietf-interfaces defined in [RFC8343].

```
<CODE BEGINS> file "ietf-nrp-device@2026-06-01.yang"
module ietf-nrp-device {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-nrp-device";
  prefix nrp-dev;

  /* Import IETF interface module */

  import ietf-interfaces {
    prefix if;
    reference
      "RFC8343: A YANG Data Model for Interface Management";
  }

  /* Import NRPs module */

  import ietf-nrp {
    prefix nrp;
    reference
      "RFCXXXX: YANG Data Models for Network Resource
        Partitions (NRPs)";
  }

  organization
    "IETF Traffic Engineering Architecture and Signaling (TEAS)
    Working Group";
  contact
    "WG Web:  <https://datatracker.ietf.org/wg/teas/>
    WG List:  <mailto:teas@ietf.org>

    Editor:   Bo Wu
              <mailto:lane.wubo@huawei.com>

    Editor:   Dhruv Dhody
              <mailto:dhruv.ietf@gmail.com>

    Editor:   Vishnu Pavan Beeram
              <mailto:vbeeram@juniper.net>

    Editor:   Tarek Saad
              <mailto:tsaad.net@gmail.com>

    Editor:   Shaofu Peng
              <mailto:peng.shaofu@zte.com.cn>";
  description
    "This YANG module defines a data model for Network Resource
    Partitions (NRPs) device configurations and states. The model
    fully conforms to the Network Management Datastore
```

Architecture (NMDA).

Copyright (c) 2026 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 (<https://www.rfc-editor.org/info/rfcXXXX>); see the RFC itself for full legal notices.";

```
// RFC Ed.: replace XXXX with actual RFC number and
// remove this note.
// RFC Ed.: update the date below with the date of RFC
// publication and remove this note.

revision 2026-06-01 {
  description
    "Initial revision.";
  reference
    "RFCXXXX: YANG Data Models for Network Resource
    Partitions (NRPs)";
}

/*
 * Grouping - NRP QoS Per-Hop Behavior (PHB) profile
 */

grouping phb-profile-ref {
  description
    "Grouping for NRP QoS Per-Hop Behavior (PHB) profile.";
  leaf phb-profile {
    type leafref {
      path "/nrp-policies/qos-profiles/phb-profile-identifier/id";
    }
  }
  description
    "PHB profile identifier, specifying the forwarding treatment
    of packets belonging to a specific NRP selector identifier,
    such as bandwidth control, congestion control
    (e.g., Section 3.4 of [RFC3644]).
    The PHB may be standard PHB, such as Assured
    Forwarding (AF), Expedited Forwarding (EF), or
    a customized local policy, such as 'High', 'Low',
```

```
        'Standard'.";
    }
}

/* NRP device configuraiton */

container nrp-policies {
  description
    "Container for nrp policies.";
  uses nrp:qos-profiles;
  list nrp-policy {
    key "name";
    unique "nrp-id";
    description
      "List of NRP policies.";
    leaf name {
      type string;
      description
        "A string that uniquely identifies the NRP policy.";
    }
    leaf nrp-id {
      type uint32;
      description
        "A 32-bit ID that uniquely identifies the NRP
        created by the enforcement of this NRP policy.";
    }
  }
  uses nrp:resource-reservation;
  uses nrp:selector-id;
  uses phb-profile-ref;
  uses nrp:igp-congruent;
  /* NRP Interface Configuration Data */
  container interfaces {
    description
      "NRP interfaces global configuration.";
    list interface {
      key "interface";
      description
        "The list of interfaces enabled for NRP.";
      leaf interface {
        type if:interface-ref;
        description
          "NRP interface name.";
      }
      uses nrp:resource-reservation;
      uses nrp:selector-id;
      uses phb-profile-ref;
    }
  }
}
```

```
    }  
  }  
}  
<CODE ENDS>
```

6. Scalability Considerations

[I-D.ietf-teas-nrp-scalability] analyzes the scalability considerations of the control plane and data plane in the NRPs deployment. This section complements some scalability considerations with the model and the possible implications on deployment or implementation.

As discussed in Section 2 of [I-D.ietf-teas-nrp-scalability], the number of Network Resource Partitions (NRPs) required depends on the deployment scenario: multi-service networks typically need around ten NRPs, whereas industrial-vertical and cloud-network expansions can require hundreds to thousands of NRPs to support correspondingly large numbers of Network Slices. Compared to a single NRP, all these numbers of NRPs pose challenges for management and operations, including NRP planning, provisioning, monitoring, and Network Slice Service provisioning and assurance based on NRPs. Section 3 of [I-D.ietf-teas-ns-ip-mpls] describes the detailed realization process of multiple NRPs.

For better scalability, the NRP YANG model design considers the following aspects:

- * Planning: A reusable "qos-profile" groups Service-Level Objectives (SLOs) once and applies them to any NRP, guaranteeing consistent QoS policy across the domain while eliminating per-NRP duplication. "filter/selection" enables multiple NRPs to share a common topology, confining per-NRP state to the differences rather than replicating the entire topology. Network operators should create new NRPs only when strict resource isolation is required, ensuring that network-wide resources remain efficiently utilized.
- * Provisioning: The network controller, not the Network Slice Controller (NSC), is responsible for the actual resource allocation of NRPs on devices. It assigns NRPs to the relevant interfaces, reserve the bandwidth, and set data-plane selector IDs. NSC conveys the intent through the NRP network model, specifically via the "nrp-policy", and the network controller then realize the requested allocation. And for data-plane selector IDs, Section 5.2 of [I-D.ietf-teas-nrp-scalability] recommends selecting a data-plane mechanism that minimizes per-NRP configuration. A dedicated IPv6 selector identifier is one example that achieves this objective.

- * Monitoring: The model supports two complementary monitoring approaches:

Per-NRP view: Operational state scoped to a single "nrp-id", letting the controller present the logical network dedicated to that NRP.

Network-wide view: An extension of the RFC 8345 topology model that exposes the mapping between the underlay network and all associated NRPs, including per-node and per-link NRP status.

7. Security Considerations

This section is modeled after the template described in Section 3.7.1 of [RFC9907].

The "ietf-nrp" and "ietf-nrp-device" YANG modules define data models that are designed to be accessed via YANG-based management protocols, such as the Network Configuration Protocol (NETCONF) [RFC6241] and RESTCONF [RFC8040]. These YANG-based management protocols (1) have to use a secure transport layer (e.g., Secure Shell (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.

There are a number of data nodes defined in this YANG model that are writable/creatable/deletable (i.e., config true, which is the default). These data nodes may be considered sensitive or vulnerable in some network environments. Write operations (e.g., edit-config) to these data nodes without proper protection can have a negative effect on network operations. The following subtrees and data nodes have particular sensitivities/vulnerabilities:

nrp-policies: Unauthorized creation or modification of NRP policies can allocate excessive network resources (bandwidth, buffers, forwarding entries), causing denial of service to other services. Implementations MUST enforce resource limits and validate quota constraints before applying NRP configurations. NETCONF access control SHOULD restrict NRP policy creation to authorized network-admin roles only.

selector-id and phb-profile: Misconfiguration of NRP selector IDs or PHB profiles may cause traffic to be forwarded on the wrong NRP, violating isolation guarantees and SLOs. Implementations MUST

validate selector ID uniqueness within the network scope and prevent overlapping PHB configurations that could create ambiguous forwarding rules.

topology: Modification of NRP topology (links, nodes, filters) could reroute traffic through unintended paths, bypassing security policies or causing congestion. Operators MUST apply access controls to topology modifications and audit all changes. The model's operational state SHOULD be monitored to detect unauthorized topology alterations.

The entries in the nodes above include the whole network configurations corresponding with the NRP, and indirectly create or modify the PE or P device configurations. Unexpected changes to these entries could lead to service disruption and/or network misbehavior.

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:

Sensitive Data Exposure: Operational state data (e.g., utilized bandwidth, delay statistics) may reveal sensitive information about network load and customer traffic patterns. Access to operational state SHOULD be restricted using NACM rules, limiting visibility to authorized monitoring systems.

8. IANA Considerations

This document registers a URI in the IETF XML registry [RFC3688]. Following the format in [RFC3688], the following registration is requested to be made:

URI: urn:ietf:params:xml:ns:yang:ietf-nrp
Registrant Contact: The IESG.
XML: N/A, the requested URI is an XML namespace.

URI: urn:ietf:params:xml:ns:yang:ietf-nrp-device
Registrant Contact: The IESG.
XML: N/A, the requested URI is an XML namespace.

This document requests to register a YANG module in the YANG Module Names registry [RFC7950].

Name: ietf-nrp
Namespace: urn:ietf:params:xml:ns:yang:ietf-nrp
Maintained by IANA: N
Prefix: nrp
Reference: RFC XXXX

Name: ietf-nrp-device
Namespace: urn:ietf:params:xml:ns:yang:ietf-nrp-device
Maintained by IANA: N
Prefix: nrp-dev
Reference: RFC XXXX

9. Acknowledgments

The authors would like to thank Krzysztof Szarkowicz, Jie Dong, Qin Wu, Yao Zhao, Zhenbing Li, Adrian Farrel, Tom Petch, Xuesong Geng, Italo Busi, and many others for their helpful comments and suggestions.

10. Contributor

The following individuals, authors of [I-D.bestbar-teas-yang-nrp-policy] and [I-D.wd-teas-nrp-yang], contributed to this consolidated document:

Xufeng Liu
IBM Corporation
Email: xufeng.liu.ietf@gmail.com

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

Daniele Ceccarelli

Bin Wen
Comcast
Email: Bin_Wen@cable.comcast.com

Ran Chen
ZTE Corporation
Email: chen.ran@zte.com.cn

Luis M. Contreras
Telefonica
Email: luismiguel.contrerasmurillo@telefonica.com

Ying Cheng
China Unicom
Email: chengying10@chinaunicom.cn

Liyan Gong
China Mobile
Email: gongliyan@chinamobile.com

11. References

11.1. Normative References

[I-D.ietf-spring-srv6-yang-base]
Raza, S. K., Rajamanickam, J., Matsushima, S., Yu, P., and X. Liu, "YANG Data Model for SRv6 Base", Work in Progress, Internet-Draft, draft-ietf-spring-srv6-yang-base-02, 14 May 2026, <<https://datatracker.ietf.org/doc/html/draft-ietf-spring-srv6-yang-base-02>>.

[I-D.ietf-teas-yang-topology-filter]
Beeram, V. P., Saad, T., Gandhi, R., and X. Liu, "YANG Data Model for Topology Filter", Work in Progress, Internet-Draft, draft-ietf-teas-yang-topology-filter-02, 19 October 2025, <<https://datatracker.ietf.org/doc/html/draft-ietf-teas-yang-topology-filter-02>>.

- [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>>.
- [RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, DOI 10.17487/RFC3688, January 2004, <<https://www.rfc-editor.org/info/rfc3688>>.
- [RFC4915] Psenak, P., Mirtorabi, S., Roy, A., Nguyen, L., and P. Pillay-Esnault, "Multi-Topology (MT) Routing in OSPF", RFC 4915, DOI 10.17487/RFC4915, June 2007, <<https://www.rfc-editor.org/info/rfc4915>>.
- [RFC5120] Przygienda, T., Shen, N., and N. Sheth, "M-ISIS: Multi Topology (MT) Routing in Intermediate System to Intermediate Systems (IS-ISs)", RFC 5120, DOI 10.17487/RFC5120, February 2008, <<https://www.rfc-editor.org/info/rfc5120>>.
- [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/info/rfc6241>>.
- [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>>.
- [RFC7951] Lhotka, L., "JSON Encoding of Data Modeled with YANG", RFC 7951, DOI 10.17487/RFC7951, August 2016, <<https://www.rfc-editor.org/info/rfc7951>>.
- [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>>.
- [RFC8294] Liu, X., Qu, Y., Lindem, A., Hopps, C., and L. Berger, "Common YANG Data Types for the Routing Area", RFC 8294, DOI 10.17487/RFC8294, December 2017, <<https://www.rfc-editor.org/info/rfc8294>>.
- [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>>.

- [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/info/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/info/rfc8342>>.
- [RFC8343] Bjorklund, M., "A YANG Data Model for Interface Management", RFC 8343, DOI 10.17487/RFC8343, March 2018, <<https://www.rfc-editor.org/info/rfc8343>>.
- [RFC8345] Clemm, A., Medved, J., Varga, R., Bahadur, N., Ananthakrishnan, H., and X. Liu, "A YANG Data Model for Network Topologies", RFC 8345, DOI 10.17487/RFC8345, March 2018, <<https://www.rfc-editor.org/info/rfc8345>>.
- [RFC8519] Jethanandani, M., Agarwal, S., Huang, L., and D. Blair, "YANG Data Model for Network Access Control Lists (ACLs)", RFC 8519, DOI 10.17487/RFC8519, March 2019, <<https://www.rfc-editor.org/info/rfc8519>>.
- [RFC8776] Saad, T., Gandhi, R., Liu, X., Beeram, V., and I. Bryskin, "Common YANG Data Types for Traffic Engineering", RFC 8776, DOI 10.17487/RFC8776, June 2020, <<https://www.rfc-editor.org/info/rfc8776>>.
- [RFC9181] Barguil, S., Gonzalez de Dios, O., Ed., Boucadair, M., Ed., and Q. Wu, "A Common YANG Data Model for Layer 2 and Layer 3 VPNs", RFC 9181, DOI 10.17487/RFC9181, February 2022, <<https://www.rfc-editor.org/info/rfc9181>>.
- [RFC9543] Farrel, A., Ed., Drake, J., Ed., Rokui, R., Homma, S., Makhijani, K., Contreras, L., and J. Tantsura, "A Framework for Network Slices in Networks Built from IETF Technologies", RFC 9543, DOI 10.17487/RFC9543, March 2024, <<https://www.rfc-editor.org/info/rfc9543>>.
- [RFC9911] Schoenwaelder, J., Ed., "Common YANG Data Types", RFC 9911, DOI 10.17487/RFC9911, December 2025, <<https://www.rfc-editor.org/info/rfc9911>>.

11.2. Informative References

[I-D.bestbar-teas-yang-nrp-policy]

Beeram, V. P., Saad, T., Wen, B., Ceccarelli, D., Peng, S., Chen, R., Contreras, L. M., and X. Liu, "YANG Data Model for Network Resource Partition Policy", Work in Progress, Internet-Draft, draft-bestbar-teas-yang-nrp-policy-03, 24 October 2022, <<https://datatracker.ietf.org/doc/html/draft-bestbar-teas-yang-nrp-policy-03>>.

[I-D.ietf-6man-enhanced-vpn-vtn-id]

Dong, J., Li, Z., Xie, C., Ma, C., and G. S. Mishra, "Carrying Network Resource (NR) related Information in IPv6 Extension Header", Work in Progress, Internet-Draft, draft-ietf-6man-enhanced-vpn-vtn-id-15, 23 March 2026, <<https://datatracker.ietf.org/doc/html/draft-ietf-6man-enhanced-vpn-vtn-id-15>>.

[I-D.ietf-lsr-isis-sr-vtn-mt]

Xie, C., Ma, C., Dong, J., and Z. Li, "Applicability of IS-IS Multi-Topology (MT) for Segment Routing based Network Resource Partition (NRP)", Work in Progress, Internet-Draft, draft-ietf-lsr-isis-sr-vtn-mt-12, 27 May 2026, <<https://datatracker.ietf.org/doc/html/draft-ietf-lsr-isis-sr-vtn-mt-12>>.

[I-D.ietf-mppls-mna-nrp-selector]

Li, T., Beeram, V. P., Drake, J., Saad, T., and I. Meilik, "MPLS Network Actions for Network Resource Partition Selector", Work in Progress, Internet-Draft, draft-ietf-mppls-mna-nrp-selector-06, 5 May 2026, <<https://datatracker.ietf.org/doc/html/draft-ietf-mppls-mna-nrp-selector-06>>.

[I-D.ietf-mppls-mna-psd-nrp-selector]

Li, Z. and J. Dong, "Carrying NRP related Information in MPLS Packets", Work in Progress, Internet-Draft, draft-ietf-mppls-mna-psd-nrp-selector-01, 2 March 2026, <<https://datatracker.ietf.org/doc/html/draft-ietf-mppls-mna-psd-nrp-selector-01>>.

[I-D.ietf-spring-sr-for-enhanced-vpn]

Dong, J., Miyasaka, T., Zhu, Y., Qin, F., and Z. Li, "Segment Routing based Network Resource Partition (NRP) for Enhanced VPN", Work in Progress, Internet-Draft, draft-ietf-spring-sr-for-enhanced-vpn-10, 15 December 2025, <<https://datatracker.ietf.org/doc/html/draft-ietf-spring-sr-for-enhanced-vpn-10>>.

[I-D.ietf-teas-nrp-scalability]

Dong, J., Li, Z., Gong, L., Yang, G., and G. S. Mishra,
"Scalability Considerations for Network Resource
Partition", Work in Progress, Internet-Draft, draft-ietf-
teas-nrp-scalability-09, 11 February 2026,
<[https://datatracker.ietf.org/doc/html/draft-ietf-teas-
nrp-scalability-09](https://datatracker.ietf.org/doc/html/draft-ietf-teas-nrp-scalability-09)>.

[I-D.ietf-teas-ns-ip-mpls]

Saad, T., Beeram, V. P., Dong, J., Halpern, J. M., and S.
Peng, "Realizing Network Slices in IP/MPLS Networks", Work
in Progress, Internet-Draft, draft-ietf-teas-ns-ip-mpls-
07, 28 February 2026,
<[https://datatracker.ietf.org/doc/html/draft-ietf-teas-ns-
ip-mpls-07](https://datatracker.ietf.org/doc/html/draft-ietf-teas-ns-ip-mpls-07)>.

[I-D.wd-teas-nrp-yang]

Wu, B., Dhody, D., Boucadair, M., Cheng, Y., and L. Gong,
"A YANG Data Model for Network Resource Partitions
(NRPs)", Work in Progress, Internet-Draft, draft-wd-teas-
nrp-yang-02, 25 September 2022,
<[https://datatracker.ietf.org/doc/html/draft-wd-teas-nrp-
yang-02](https://datatracker.ietf.org/doc/html/draft-wd-teas-nrp-yang-02)>.

[RFC3644] Snir, Y., Ramberg, Y., Strassner, J., Cohen, R., and B.
Moore, "Policy Quality of Service (QoS) Information
Model", RFC 3644, DOI 10.17487/RFC3644, November 2003,
<<https://www.rfc-editor.org/info/rfc3644>>.

[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/info/rfc4252>>.

[RFC7224] Bjorklund, M., "IANA Interface Type YANG Module",
RFC 7224, DOI 10.17487/RFC7224, May 2014,
<<https://www.rfc-editor.org/info/rfc7224>>.

[RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF
Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017,
<<https://www.rfc-editor.org/info/rfc8040>>.

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

[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/info/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/info/rfc9000>>.
- [RFC9350] Psenak, P., Ed., Hegde, S., Filsfils, C., Talaulikar, K., and A. Gulko, "IGP Flexible Algorithm", RFC 9350, DOI 10.17487/RFC9350, February 2023, <<https://www.rfc-editor.org/info/rfc9350>>.
- [RFC9907] Bierman, A., Boucadair, M., Ed., and Q. Wu, "Guidelines for Authors and Reviewers of Documents Containing YANG Data Models", BCP 216, RFC 9907, DOI 10.17487/RFC9907, March 2026, <<https://www.rfc-editor.org/info/rfc9907>>.

Appendix A. Examples

This section provides example instance data trees in JSON encoding [RFC7951] for the NRP YANG models defined in this document. The examples illustrate three typical scenarios:

- * Example 1: A network-level NRP policy with Hybrid Partition mode for low-latency services using SRv6 and Flex-Algo.
- * Example 2: A network-level NRP policy with Control-Plane Partition mode for high-bandwidth services using topology filter.
- * Example 3: A device-level NRP Policies with per-Interface configuration.

Figure 10 shows an example of three nodes NRPs network: D1,D2, and D3.

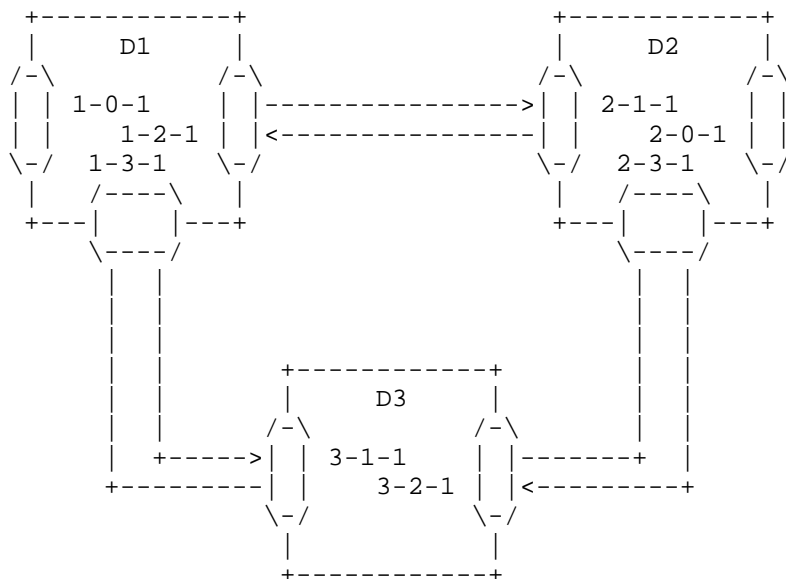


Figure 10: Example of NRPs Network

A.1. Network-level NRP Policy with Hybrid Partition Example

Figure 11 shows an example of a hybrid NRP named "NRP1-Low-Latency". It uses an SRv6 SID as the traffic selector and associates the NRP with an IGP Flexible Algorithm (Algo 128) to compute low-latency paths.

```

===== NOTE: '\ ' line wrapping per RFC 8792 =====
{
  "ietf-network:networks": {
    "ietf-nrp:nrp-policies": {
      "qos-profiles": {
        "phb-profile-identifier": [
          {
            "id": "high-bandwidth"
          },
          {
            "id": "low-latency"
          },
          {
            "id": "standard"
          }
        ]
      },
      "nrp-policy": [
        {
          "name": "NRP1-Low-Latency",
          "nrp-id": 100,
          "mode": "ietf-nrp:hybrid-plane-partition",
          "resource-reservation": {
            "maximum-bandwidth": "1000000000"
          },
          "selector-id": {
            "ipv6": {
              "srv6-sid": [
                "2001:db8:100:0:0:0:0:0/128"
              ]
            }
          },
          "phb-profile": "low-latency",
          "topology": {
            "igp-congruent": {
              "algo-id": 128,
              "sharing": false
            }
          }
        }
      ]
    }
  }
}

```

Figure 11: Example of a Message Body to Create a Hybrid NRP Policy

A.2. Network-Level NRP Policy with Control-Plane Partition Example

Figure 12 shows a NRP policy named "NRP2-High-Bandwidth" with control plane partition mode. It relies on a topology filter to select a subset of network links capable of supporting large bandwidth reservations. The topology filters is configured using "ietf-topology-filter" defined in [I-D.ietf-teas-yang-topology-filter].

```
{
  "ietf-network:networks": {
    "ietf-topology-filter:topology-filters": {
      "topology-filter": [
        {
          "name": "example-topo-filter-1",
          "topology-ref": {
            "igp-domain-identifier": {
              "protocol-id": "ospfv3",
              "instance-id": 1,
              "division-id": 0,
              "algo-id": 128,
              "mt-id": 0
            }
          },
          "include-any": {
            "node-prefix": [
              "2001:db8:d1::/48",
              "2001:db8:d2::/48",
              "2001:db8:d3::/48"
            ]
          }
        }
      ]
    },
    "ietf-nrp:nrp-policies": {
      "qos-profiles": {
        "phb-profile-identifier": [
          {
            "id": "high-bandwidth"
          },
          {
            "id": "low-latency"
          },
          {
            "id": "standard"
          }
        ]
      },
      "nrp-policy": [
```

```

{
  "name": "NRP2-High-Bandwidth",
  "nrp-id": 200,
  "mode": "control-plane-partition",
  "resource-reservation": {
    "maximum-bandwidth": "10000000000"
  },
  "phb-profile": "high-bandwidth",
  "topology": {
    "filters": {
      "filter": [
        {
          "filter-ref": "example-topo-filter-1",
          "resource-reservation": {
            "maximum-bandwidth": "10000000000"
          }
        }
      ]
    }
  }
}

```

Figure 12: Example of a Message Body to Create Network-Level NRP Policy with Control-Plane Partition

A.3. Device-Level NRP Policy Example

Figure 13 shows the configuration of two NRP policies on a router using the "ietf-nrp-device" module. The interfaces of the router are configured using the "ietf-interfaces" module [RFC8343], and the interface type is specified using the "iana-if-type" module [RFC7224]. In this example, NRP-Premium-Service provides high-priority service on interfaces GE0/0/0 and GE0/0/1, while NRP-Standard-Service provides standard service on interface GE0/0/2.

```

{
  "ietf-interfaces:interfaces": {
    "interface": [
      {
        "name": "GigabitEthernet0/0/0/0",
        "type": "iana-if-type:ethernetCsmacd",
        "enabled": true
      },
      {

```

```
    "name": "GigabitEthernet0/0/0/1",
    "type": "iana-if-type:ethernetCsmacd",
    "enabled": true
  },
  {
    "name": "GigabitEthernet0/0/0/2",
    "type": "iana-if-type:ethernetCsmacd",
    "enabled": true
  }
]
},
"ietf-nrp-device:nrp-policies": {
  "qos-profiles": {
    "phb-profile-identifier": [
      {
        "id": "High-Priority-PHB"
      },
      {
        "id": "Standard-PHB"
      }
    ]
  },
  "nrp-policy": [
    {
      "name": "NRP-Premium-Service",
      "nrp-id": 100,
      "resource-reservation": {
        "maximum-bandwidth-percent": 20
      },
      "selector-id": {
        "ipv6": {
          "ipv6-hbh-eh": 100
        }
      },
      "phb-profile": "High-Priority-PHB",
      "igp-congruent": {
        "algo-id": 128,
        "sharing": false
      },
      "interfaces": {
        "interface": [
          {
            "interface": "GigabitEthernet0/0/0/0",
            "resource-reservation": {
              "maximum-bandwidth-percent": 40
            },
            "selector-id": {
              "ipv6": {
```

```

        "ipv6-hbh-eh": 101
      }
    },
    "phb-profile": "High-Priority-PHB"
  },
  {
    "interface": "GigabitEthernet0/0/0/1",
    "resource-reservation": {
      "maximum-bandwidth-percent": 30
    },
    "selector-id": {
      "ipv6": {
        "ipv6-hbh-eh": 102
      }
    },
    "phb-profile": "High-Priority-PHB"
  }
]
}
},
{
  "name": "NRP-Standard-Service",
  "nrp-id": 300,
  "resource-reservation": {
    "maximum-bandwidth-percent": 50
  },
  "selector-id": {
    "ipv6": {
      "ipv6-hbh-eh": 300
    }
  },
  "phb-profile": "Standard-PHB",
  "interfaces": {
    "interface": [
      {
        "interface": "GigabitEthernet0/0/0/2",
        "resource-reservation": {
          "maximum-bandwidth-percent": 50
        },
        "selector-id": {
          "ipv6": {
            "ipv6-hbh-eh": 301
          }
        },
        "phb-profile": "Standard-PHB"
      }
    ]
  }
}
}

```

```

    }
  ]
}

```

Figure 13: Example of a Message Body to Create Device NRP Policy

Appendix B. NRPs YANG Module Tree

Figure 14 shows the full tree diagram of the NRPs YANG model defined in module "ietf-nrp.yang".

```
module: ietf-nrp
```

```

augment /nw:networks:
  +--rw nrp-policies
    +--rw qos-profiles
      | +--rw phb-profile-identifier* [id]
      | +--rw id string
    +--rw nrp-policy* [name]
      +--rw name string
      +--rw nrp-id uint32
      +--rw mode identityref
      +--rw resource-reservation
        | +--rw (max-bw-type)?
        | | +--:(bw-value)
        | | | +--rw maximum-bandwidth? uint64
        | | +--:(bw-percentage)
        | | | +--rw maximum-bandwidth-percent?
        | | | | rt-types:percentage
      +--rw selector-id
        | +--rw ipv4-derived {ipv4-selector}?
        | | +--rw destination-prefix* inet:ipv4-prefix
        | +--rw ipv6
        | | {ipv6-hbh-selector or srv6-selector or
        | | | ipv6-destination-selector}?
        | | | +--rw (selector-type)?
        | | | | +--:(dedicated) {ipv6-hbh-selector}?
        | | | | | +--rw ipv6-hbh-eh? uint32
        | | | | +--:(srv6-derived) {srv6-selector}?
        | | | | | +--rw srv6-sid* srv6-types:srv6-sid
        | | | | +--:(ipv6-destination-derived)
        | | | | | {ipv6-destination-selector}?
        | | | | | +--rw destination-prefix* inet:ipv6-prefix
      +--rw mpls
        | {mpls-mna-selector or mpls-post-stack-selector}?
        +--rw (mpls-encapsulation)?
        | +--:(mna-in-stack) {mpls-mna-selector}?

```

```

| | | | +--rw in-stack-identifier?      uint32
| | | | +--rw identifier-format?        enumeration
| | | | +---:(post-stack) {mpls-post-stack-selector}?
| | | | | +--rw post-stack-identifier?  uint32
| | | +--rw acl-ref*                    acl-ref {acl-selector}?
+--rw phb-profile?                      leafref
+--rw topology
  +--rw igp-congruent!
  | +--rw multi-topology-id?            uint32
  | +--rw algo-id?                      uint32
  | +--rw sharing?                      boolean
+--rw (topology-type)?
  +---:(selection)
  | +--rw select
  | | +--rw topology-group* [group-id]
  | | | +--rw group-id                  string
  | | | +--rw network-ref?
  | | | | -> /nw:networks/network/network-id
  | | | +--rw link* [link-ref]
  | | | | +--rw link-ref                leafref
  | | | +--rw resource-reservation
  | | | | +--rw (max-bw-type)?
  | | | | | +---:(bw-value)
  | | | | | | +--rw maximum-bandwidth?
  | | | | | | | uint64
  | | | | | | +---:(bw-percentage)
  | | | | | | | +--rw maximum-bandwidth-percent?
  | | | | | | | | rt-types:percentage
  | | | | +--rw link-partition-type?    identityref
  | | | +--rw phb-profile?              leafref
  +---:(filter)
  | +--rw filters
  | | +--rw filter* [filter-ref]
  | | | +--rw filter-ref
  | | | | topo-filter-ref
  | | | +--rw resource-reservation
  | | | | +--rw (max-bw-type)?
  | | | | | +---:(bw-value)
  | | | | | | +--rw maximum-bandwidth?
  | | | | | | | uint64
  | | | | | | +---:(bw-percentage)
  | | | | | | | +--rw maximum-bandwidth-percent?
  | | | | | | | | rt-types:percentage
  | | | +--rw selector-id
  | | | | +--rw ipv4-derived {ipv4-selector}?
  | | | | | +--rw destination-prefix*
  | | | | | | inet:ipv4-prefix
  | | | +--rw ipv6

```

```

ipv6-destination-selector}?
|
| {ipv6-hbh-selector or srv6-selector or
|
| +--rw (selector-type)?
|   +--:(dedicated)
|     {ipv6-hbh-selector}?
|     +--rw ipv6-hbh-eh?
|         uint32
|   +--:(srv6-derived) {srv6-selector}?
|     +--rw srv6-sid*
|         srv6-types:srv6-sid
|   +--:(ipv6-destination-derived)
|     {ipv6-destination-selector}?
|     +--rw destination-prefix*
|         inet:ipv6-prefix
|
| +--rw mpls
|   {mpls-mna-selector or mpls-post-stack-selector}?
|   +--rw (mpls-encapsulation)?
|     +--:(mna-in-stack)
|       {mpls-mna-selector}?
|       +--rw in-stack-identifier?
|           uint32
|       +--rw identifier-format?
|           enumeration
|     +--:(post-stack)
|       {mpls-post-stack-selector}?
|       +--rw post-stack-identifier?
|           uint32
|
| +--rw acl-ref*          acl-ref
|   {acl-selector}?
|
+--rw phb-profile?      leafref

augment /nw:networks/nw:network/nw:network-types:
+--rw nrp!

augment /nw:networks/nw:network/nw:node:
+--ro (nrp-attachment-type)?
+--:(single-nrp)
| +--ro nrp-node-attributes
|   +--ro selector-id
|   +--ro srv6?          srv6-types:srv6-sid
|   +--ro mpls-nrp-selector?  uint32
+--:(multi-nrp)
+--ro node-nrp-list
+--ro nrp* [nrp-id]
+--ro nrp-id            uint32
+--ro nrp-node-attributes
+--ro selector-id
+--ro srv6?          srv6-types:srv6-sid
+--ro mpls-nrp-selector?  uint32

augment /nw:networks/nw:network/nt:link:

```

```

+--ro (nrp-attachment-type)?
+--:(single-nrp)
|   +--ro nrp-link-attributes
|   |   +--ro link-partition-type?    identityref
|   |   +--ro bandwidth-value?        uint64
|   |   +--ro selector-id
|   |   |   +--ro srv6?                srv6-types:srv6-sid
|   |   |   +--ro mpls-nrp-selector?   uint32
|   |   +--ro statistics
|   |   |   +--ro status
|   |   |   |   +--ro admin-status
|   |   |   |   |   +--ro status?        identityref
|   |   |   |   |   +--ro last-change?   yang:date-and-time
|   |   |   |   +--ro oper-status
|   |   |   |   |   +--ro status?        identityref
|   |   |   |   |   +--ro last-change?   yang:date-and-time
|   |   +--ro one-way-available-bandwidth?  uint64
|   |   +--ro one-way-utilized-bandwidth?    uint64
|   |   +--ro one-way-min-delay?             uint32
|   |   +--ro one-way-max-delay?             uint32
|   |   +--ro one-way-delay-variation?       uint32
|   |   +--ro one-way-packet-loss?           decimal64
+--:(multi-nrp)
+--ro link-nrp-list
+--ro nrp* [nrp-id]
+--ro nrp-id                               uint32
+--ro nrp-link-attributes
+--ro link-partition-type?    identityref
+--ro bandwidth-value?        uint64
+--ro selector-id
|   +--ro srv6?                srv6-types:srv6-sid
|   +--ro mpls-nrp-selector?   uint32
+--ro statistics
+--ro status
|   +--ro admin-status
|   |   +--ro status?        identityref
|   |   +--ro last-change?   yang:date-and-time
|   +--ro oper-status
|   |   +--ro status?        identityref
|   |   +--ro last-change?   yang:date-and-time
+--ro one-way-available-bandwidth?  uint64
+--ro one-way-utilized-bandwidth?    uint64
+--ro one-way-min-delay?             uint32
+--ro one-way-max-delay?             uint32
+--ro one-way-delay-variation?       uint32
+--ro one-way-packet-loss?           decimal64

```

Figure 14

Appendix C. NRPs Device YANG Module Tree

Figure 15 shows the full tree diagram of the NRPs device YANG model defined in module "ietf-nrp-device.yang".

```

module: ietf-nrp-device
  +--rw nrp-policies
    +--rw qos-profiles
      | +--rw phb-profile-identifier* [id]
      |   +--rw id string
    +--rw nrp-policy* [name]
      +--rw name string
      +--rw nrp-id? uint32
      +--rw resource-reservation
        | +--rw (max-bw-type)?
        |   +--:(bw-value)
        |     | +--rw maximum-bandwidth? uint64
        |     +--:(bw-percentage)
        |       +--rw maximum-bandwidth-percent?
        |         rt-types:percentage
      +--rw selector-id
        | +--rw ipv4-derived {ipv4-selector}?
        |   +--rw destination-prefix* inet:ipv4-prefix
        +--rw ipv6
          | {ipv6-hbh-selector or srv6-selector or
          |   ipv6-destination-selector}?
          |   +--rw (selector-type)?
          |     +--:(dedicated) {ipv6-hbh-selector}?
          |       | +--rw ipv6-hbh-eh? uint32
          |     +--:(srv6-derived) {srv6-selector}?
          |       | +--rw srv6-sid* srv6-types:srv6-sid
          |     +--:(ipv6-destination-derived)
          |       {ipv6-destination-selector}?
          |       +--rw destination-prefix* inet:ipv6-prefix
          +--rw mpls
            | {mpls-mna-selector or mpls-post-stack-selector}?
            +--rw (mpls-encapsulation)?
            |   +--:(mna-in-stack) {mpls-mna-selector}?
            |     | +--rw in-stack-identifier? uint32
            |     | +--rw identifier-format? enumeration
            |   +--:(post-stack) {mpls-post-stack-selector}?
            |     +--rw post-stack-identifier? uint32
            +--rw acl-ref* acl-ref {acl-selector}?
          +--rw phb-profile? leafref
        +--rw igp-congruent!
          | +--rw multi-topology-id? uint32
          | +--rw algo-id? uint32
          +--rw sharing? boolean

```

```

+--rw interfaces
  +--rw interface* [interface]
    +--rw interface          if:interface-ref
    +--rw resource-reservation
      | +--rw (max-bw-type)?
      |   +--:(bw-value)
      |   | +--rw maximum-bandwidth?          uint64
      |   +--:(bw-percentage)
      |   | +--rw maximum-bandwidth-percent?
      |   |   rt-types:percentage
      | +--rw selector-id
      |   +--rw ipv4-derived {ipv4-selector}?
      |   | +--rw destination-prefix*    inet:ipv4-prefix
      |   +--rw ipv6
      |   | {ipv6-hbh-selector or srv6-selector or
      |   |   ipv6-destination-selector}?
      |   |   +--rw (selector-type)?
      |   |   +--:(dedicated) {ipv6-hbh-selector}?
      |   |   | +--rw ipv6-hbh-eh?          uint32
      |   |   +--:(srv6-derived) {srv6-selector}?
      |   |   | +--rw srv6-sid*
      |   |   |   srv6-types:srv6-sid
      |   |   +--:(ipv6-destination-derived)
      |   |   | {ipv6-destination-selector}?
      |   |   | +--rw destination-prefix*
      |   |   |   inet:ipv6-prefix
      |   +--rw mpls
      |   | {mpls-mna-selector or mpls-post-stack-selector}?
      |   | +--rw (mpls-encapsulation)?
      |   |   +--:(mna-in-stack) {mpls-mna-selector}?
      |   |   | +--rw in-stack-identifier?    uint32
      |   |   | +--rw identifier-format?      enumeration
      |   |   +--:(post-stack) {mpls-post-stack-selector}?
      |   |   | +--rw post-stack-identifier?  uint32
      |   +--rw acl-ref*          acl-ref {acl-selector}?
      +--rw phb-profile?          leafref

```

Figure 15

Authors' Addresses

Bo Wu
 Huawei Technologies
 101 Software Avenue, Yuhua District
 Nanjing
 Jiangsu, 210012
 China
 Email: lana.wubo@huawei.com

Dhruv Dhody
Huawei Technologies
Divyashree Techno Park
Bangalore 560066
Karnataka
India
Email: dhruv.ietf@gmail.com

Vishnu Pavan Beeram
Juniper Networks
Email: vbeeram@juniper.net

Tarek Saad
Cisco Systems
Email: tsaad.net@gmail.com

Shaofu Peng
ZTE Corporation
Email: peng.shaofu@zte.com.cn