



Nokia Service Router Linux
7215 Interconnect System
7220 Interconnect Router
7250 Interconnect Router
7730 Service Interconnect Router
Release 24.10

NetOps Development Kit Reference

3HE 20955 AAAA TQZZA
Edition: 01
November 2024

Nokia is committed to diversity and inclusion. We are continuously reviewing our customer documentation and consulting with standards bodies to ensure that terminology is inclusive and aligned with the industry. Our future customer documentation will be updated accordingly.

This document includes Nokia proprietary and confidential information, which may not be distributed or disclosed to any third parties without the prior written consent of Nokia.

This document is intended for use by Nokia's customers ("You"/"Your") in connection with a product purchased or licensed from any company within Nokia Group of Companies. Use this document as agreed. You agree to notify Nokia of any errors you may find in this document; however, should you elect to use this document for any purpose(s) for which it is not intended, You understand and warrant that any determinations You may make or actions You may take will be based upon Your independent judgment and analysis of the content of this document.

Nokia reserves the right to make changes to this document without notice. At all times, the controlling version is the one available on Nokia's site.

No part of this document may be modified.

NO WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY OF AVAILABILITY, ACCURACY, RELIABILITY, TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE, IS MADE IN RELATION TO THE CONTENT OF THIS DOCUMENT. IN NO EVENT WILL NOKIA BE LIABLE FOR ANY DAMAGES, INCLUDING BUT NOT LIMITED TO SPECIAL, DIRECT, INDIRECT, INCIDENTAL OR CONSEQUENTIAL OR ANY LOSSES, SUCH AS BUT NOT LIMITED TO LOSS OF PROFIT, REVENUE, BUSINESS INTERRUPTION, BUSINESS OPPORTUNITY OR DATA THAT MAY ARISE FROM THE USE OF THIS DOCUMENT OR THE INFORMATION IN IT, EVEN IN THE CASE OF ERRORS IN OR OMISSIONS FROM THIS DOCUMENT OR ITS CONTENT.

Copyright and trademark: Nokia is a registered trademark of Nokia Corporation. Other product names mentioned in this document may be trademarks of their respective owners.

The registered trademark Linux® is used pursuant to a sublicense from the Linux Foundation, the exclusive licensee of Linus Torvalds, owner of the mark on a worldwide basis.

© 2024 Nokia.

Table of contents

| | | |
|----------|---|-----------|
| 1 | About this guide..... | 7 |
| 1.1 | Precautionary and information messages..... | 7 |
| 1.2 | Conventions..... | 7 |
| 2 | What's new..... | 9 |
| 3 | Introduction..... | 10 |
| 3.1 | Datastore..... | 10 |
| 3.2 | gRPC..... | 10 |
| 3.3 | Protocol buffers..... | 10 |
| 4 | Protocol Documentation..... | 12 |
| 4.1 | appid_service.proto..... | 12 |
| 4.1.1 | AppIdentData..... | 12 |
| 4.1.2 | AppIdentKey..... | 12 |
| 4.1.3 | AppIdentNotification..... | 12 |
| 4.1.4 | AppIdentSubscriptionRequest..... | 13 |
| 4.2 | bfd_service.proto..... | 13 |
| 4.2.1 | BfdmgrGeneralSessionDataPb.BfdmgrSessionSubType..... | 13 |
| 4.2.2 | BfdmgrGeneralSessionKeyPb.SbfdechoKey.SrpolicyUserType..... | 13 |
| 4.2.3 | BfdmgrSessionStatus..... | 14 |
| 4.2.4 | BfdmgrSessionType..... | 14 |
| 4.2.5 | BfdSessionNotification..... | 14 |
| 4.2.6 | BfdSessionSubscriptionRequest..... | 15 |
| 4.2.7 | BfdmgrGeneralSessionDataPb..... | 15 |
| 4.2.8 | BfdmgrGeneralSessionKeyPb..... | 15 |
| 4.2.9 | BfdmgrGeneralSessionKeyPb.MicrobfdKey..... | 16 |
| 4.2.10 | BfdmgrGeneralSessionKeyPb.P2pKey..... | 16 |
| 4.2.11 | BfdmgrGeneralSessionKeyPb.SbfdechoKey..... | 16 |
| 4.3 | config_service.proto..... | 17 |
| 4.3.1 | AcknowledgeConfigRequest..... | 17 |
| 4.3.2 | AcknowledgeConfigRequestInfo..... | 17 |
| 4.3.3 | AcknowledgeConfigResponse..... | 17 |
| 4.3.4 | ConfigData..... | 18 |

| | | |
|--------|---|----|
| 4.3.5 | ConfigKey..... | 18 |
| 4.3.6 | ConfigNotification..... | 18 |
| 4.3.7 | ConfigSubscriptionRequest..... | 19 |
| 4.3.8 | SdkMgrConfigService..... | 19 |
| 4.4 | interface_service.proto..... | 19 |
| 4.4.1 | InterfaceData..... | 19 |
| 4.4.2 | InterfaceKey..... | 20 |
| 4.4.3 | InterfaceNotification..... | 20 |
| 4.4.4 | InterfaceSubscriptionRequest..... | 21 |
| 4.5 | lldp_service.proto..... | 21 |
| 4.5.1 | LldpNeighborDataPb.PortSubType..... | 21 |
| 4.5.2 | LldpNeighborKeyPb.ChassisIdType..... | 21 |
| 4.5.3 | LldpNeighborDataPb..... | 22 |
| 4.5.4 | LldpNeighborKeyPb..... | 22 |
| 4.5.5 | LldpNeighborNotification..... | 23 |
| 4.5.6 | LldpNeighborSubscriptionRequest..... | 23 |
| 4.6 | networkinstance_service.proto..... | 23 |
| 4.6.1 | NetworkInstanceData.NetInstType..... | 23 |
| 4.6.2 | NetworkInstanceData..... | 24 |
| 4.6.3 | NetworkInstanceKey..... | 24 |
| 4.6.4 | NetworkInstanceNotification..... | 24 |
| 4.6.5 | NetworkInstanceSubscriptionRequest..... | 25 |
| 4.7 | nexthop_group_service.proto..... | 25 |
| 4.7.1 | NextHop.ResolutionType..... | 25 |
| 4.7.2 | NextHop.ResolveToType..... | 25 |
| 4.7.3 | MplsNextHop..... | 25 |
| 4.7.4 | NextHop..... | 26 |
| 4.7.5 | NextHopGroup..... | 26 |
| 4.7.6 | NextHopGroupDeleteRequest..... | 26 |
| 4.7.7 | NextHopGroupDeleteResponse..... | 26 |
| 4.7.8 | NextHopGroupInfo..... | 27 |
| 4.7.9 | NextHopGroupKey..... | 27 |
| 4.7.10 | NextHopGroupNotification..... | 27 |
| 4.7.11 | NextHopGroupRequest..... | 27 |
| 4.7.12 | NextHopGroupResponse..... | 28 |
| 4.7.13 | NextHopGroupSubscriptionRequest..... | 28 |

| | | |
|--------|---------------------------------|----|
| 4.7.14 | SdkMgrNextHopGroupService..... | 28 |
| 4.8 | route_service.proto..... | 29 |
| 4.8.1 | IpRouteNotification..... | 29 |
| 4.8.2 | IpRouteSubscriptionRequest..... | 29 |
| 4.8.3 | RouteAddRequest..... | 29 |
| 4.8.4 | RouteAddResponse..... | 29 |
| 4.8.5 | RouteDeleteRequest..... | 30 |
| 4.8.6 | RouteDeleteResponse..... | 30 |
| 4.8.7 | RouteInfo..... | 30 |
| 4.8.8 | RouteKeyPb..... | 30 |
| 4.8.9 | RoutePb..... | 31 |
| 4.8.10 | SdkMgrRouteService..... | 31 |
| 4.9 | sdk_common.proto..... | 32 |
| 4.9.1 | IfEthernetDuplexModeType..... | 32 |
| 4.9.2 | IfEthernetPortSpeedType..... | 32 |
| 4.9.3 | IfLoopbackModeType..... | 33 |
| 4.9.4 | IfMgrIfType..... | 33 |
| 4.9.5 | IfOperDownReason..... | 34 |
| 4.9.6 | IfOperStateType..... | 36 |
| 4.9.7 | IfTransceiverFecType..... | 36 |
| 4.9.8 | IpAddressState..... | 37 |
| 4.9.9 | PortOperDownReason..... | 37 |
| 4.9.10 | PortWarningReason..... | 38 |
| 4.9.11 | SdkMgrOperation..... | 39 |
| 4.9.12 | SdkMgrStatus..... | 39 |
| 4.9.13 | AgentReply..... | 40 |
| 4.9.14 | EvpnEthSegIdPb..... | 40 |
| 4.9.15 | GlobalIfId..... | 40 |
| 4.9.16 | IpAddrPrefLenPb..... | 40 |
| 4.9.17 | IpAddressPb..... | 40 |
| 4.9.18 | IpInterfaceAddrPrefixPb..... | 41 |
| 4.9.19 | MacAddressPb..... | 41 |
| 4.9.20 | MplsLabel..... | 41 |
| 4.9.21 | NetInstanceId..... | 41 |
| 4.9.22 | PortIdPb..... | 41 |
| 4.9.23 | SyncRequest..... | 42 |

| | | |
|---------|--|----|
| 4.9.24 | SyncResponse..... | 42 |
| 4.10 | sdk_service.proto..... | 42 |
| 4.10.1 | NotificationRegisterRequest.Operation..... | 42 |
| 4.10.2 | AgentRegistrationRequest..... | 42 |
| 4.10.3 | AgentRegistrationResponse..... | 43 |
| 4.10.4 | AppIdRequest..... | 44 |
| 4.10.5 | AppIdResponse..... | 44 |
| 4.10.6 | KeepAliveRequest..... | 44 |
| 4.10.7 | KeepAliveResponse..... | 44 |
| 4.10.8 | Notification..... | 44 |
| 4.10.9 | NotificationQueryRequest..... | 45 |
| 4.10.10 | NotificationQueryResponse..... | 45 |
| 4.10.11 | NotificationQuerySubscription..... | 45 |
| 4.10.12 | NotificationRegisterRequest..... | 46 |
| 4.10.13 | NotificationRegisterResponse..... | 46 |
| 4.10.14 | NotificationStreamRequest..... | 47 |
| 4.10.15 | NotificationStreamResponse..... | 47 |
| 4.10.16 | SdkMgrService..... | 47 |
| 4.10.17 | SdkNotificationService..... | 48 |
| 4.11 | telemetry_service.proto..... | 48 |
| 4.11.1 | TelemetryData..... | 48 |
| 4.11.2 | TelemetryDeleteRequest..... | 49 |
| 4.11.3 | TelemetryDeleteResponse..... | 49 |
| 4.11.4 | TelemetryInfo..... | 49 |
| 4.11.5 | TelemetryKey..... | 49 |
| 4.11.6 | TelemetryUpdateRequest..... | 50 |
| 4.11.7 | TelemetryUpdateResponse..... | 50 |
| 4.11.8 | SdkMgrTelemetryService..... | 50 |
| 4.12 | Scalar Value Types..... | 50 |
| 4.12.1 | Scalar..... | 50 |

1 About this guide

The NetOps Development Kit (NDK) allows operators to program high-performance, integrated agents that run alongside the Nokia Service Router Linux (SR Linux). This document provides programming gRPC APIs used with the NDK.

This document is intended for users who plan to program high-performance, integrated agents for SR Linux.

**Note:**

This manual covers the current release and may also contain some content that will be released in later maintenance loads. See the *SR Linux Release Notes* for information on features supported in each load.

Configuration and command outputs shown in this guide are examples only; actual displays may differ depending on supported functionality and user configuration.

1.1 Precautionary and information messages

The following are information symbols used in the documentation.



DANGER: Danger warns that the described activity or situation may result in serious personal injury or death. An electric shock hazard could exist. Before you begin work on this equipment, be aware of hazards involving electrical circuitry, be familiar with networking environments, and implement accident prevention procedures.



WARNING: Warning indicates that the described activity or situation may, or will, cause equipment damage, serious performance problems, or loss of data.



Caution: Caution indicates that the described activity or situation may reduce your component or system performance.



Note: Note provides additional operational information.



Tip: Tip provides suggestions for use or best practices.

1.2 Conventions

Nokia SR Linux documentation uses the following command conventions.

- **Bold** indicates a command that the user must enter.
- Input and output examples are displayed in Courier text.
- An open right angle bracket indicates a progression of menu choices or simple command sequence (often selected from a user interface). Example: **start > connect to**

- A vertical bar (|) indicates a mutually exclusive argument.
- Square brackets ([]) indicate optional elements.
- Braces ({ }) indicate a required choice. When braces are contained within square brackets, they indicate a required choice within an optional element.
- *Italic* indicates a variable.

Generic IP addresses are used in examples. Replace these with the appropriate IP addresses used in the system.

2 What's new

| Topic | Location |
|--|------------------------------------|
| IfOperDownReason updated with two new reasons: IF_OPER_DOWN_STORM_CONTROL_ACTION IF_OPER_DOWN_UNSUPPORTED_NUM_CHANNELS_FOR_SPEED | IfOperDownReason |
| PortOperDownReason updated with new reason: PORT_OPER_DOWN_UNSUPPORTED_BREAKOUT | PortOperDownReason |

3 Introduction

SR Linux provides a NetOps Development Kit (NDK), with a suite of libraries to assist operators with developing agents that run alongside SR Linux applications.

Agents built with the gRPC NDK function similar to other applications provided with SR Linux. SR Linux applications share state details with each other using a publish/ subscribe (pub/sub) architecture. Agents have their own table space within the IDB and can subscribe and receive a notification to events occurring on the device, or create their own table space and publish data to it. This data can be read by other applications within SR Linux, allowing route modifications by publishing routes to the IDB for selection by the FIB manager.

3.1 Datastore

The gRPC NDK allows you to add your own configuration to the system in a non-persistent manner. An NDK-added configuration is considered short-term, and its state is bound to the state of the agent. If an agent fails, the configuration added by the agent using the NDK is removed.

Agents can also add configurations through normal APIs (gNMI/JSON-RPC/CLI). This configuration persists across an agent failure, and the only way to remove it is to overwrite it with a commit command. The short-term datastore is used for agent route injection, while traditional methods for configuring the device are persistent.

3.2 gRPC

SR Linux uses gRPC for inter-process communication. gRPC is a client application that directly calls methods on a server application on a different machine as if it was a local object. The supported external APIs (CLI, gNMI, and JSON-RPC) communicate with the SR Linux and retrieve state information using gRPC.

On the server side, the server implements the interface and runs a gRPC server to handle client calls. On the client side, the client has a stub (or client) that provides the same methods as the server.

gRPC clients and servers can run and talk to each other in a number of environments and can be written in any supported gRPC language. Clients can be created in Go, Python, Ruby, or any other language with gRPC support.

3.3 Protocol buffers

SR Linux's gRPC NDK uses protocol buffers. Protocol buffers are automated mechanisms for serializing structured data. You define how you want your data to be structured once, then you can use special generated source code to easily write and read your structured data to and from a variety of data streams using a variety of languages. You can also update a data structure without breaking deployed programs that are compiled against an old format.

When working with protocol buffers, structure is defined for serialized data in a proto file (a regular text file with a .proto extension). Each protocol buffer message is a small logical record of information containing a series of name-value pairs. Each message type has one or more uniquely numbered fields, and each field has a name and value type.

Messages also have optional arguments that specify if fields are optional, required, or repeated. New fields can be added to message formats without breaking backwards compatibility, and old binaries ignore any new fields when parsing the message. This allows the gRPC NDK to evolve over time without impacting current deployments.

Once the data structure is specified, a protocol buffer compiler (protoc) generates data access classes from the proto definition. These provide simple accessors for each field (like `ConfigData()` and `set_ConfigData()`) and methods to serialize and parse the complete structure to and from raw bytes.

4 Protocol Documentation

4.1 appid_service.proto

4.1.1 AppIdentData

Represents appid data.

Table 1: AppIdentData

| Field | Type | Label | Description |
|--------------|------------------------|-------|-------------------------|
| name | string | | Application name |
| author | string | | Author name |
| is_connected | bool | | Connected to IDB or not |
| version | string | | Version string |

4.1.2 AppIdentKey

Represents appid key.

Table 2: AppIdentKey

| Field | Type | Label | Description |
|-------|------------------------|-------|----------------|
| id | uint32 | | Application id |

4.1.3 AppIdentNotification

Represents appid notification.

Table 3: AppIdentNotification

| Field | Type | Label | Description |
|-------|---------------------------------|-------|---|
| op | SdkMgrOperation | | Operation such as create, delete, or update |
| key | AppIdentKey | | AppIdent key |

| Field | Type | Label | Description |
|-------|------------------------------|-------|---------------|
| data | AppldentData | | Appldent data |

4.1.4 AppldentSubscriptionRequest

Represents appid subscription request.

Table 4: *AppldentSubscriptionRequest*

| Field | Type | Label | Description |
|-------|-----------------------------|-------|-----------------------------|
| key | AppldentKey | | Optional, to filter on name |

4.2 bfd_service.proto

4.2.1 BfdmgrGeneralSessionDataPb.BfdmgrSessionSubType

Represents BFD session subtype.

Table 5: *BfdmgrGeneralSessionDataPb.BfdmgrSessionSubType*

| Name | Number | Description |
|-----------------------------|--------|-------------------------|
| SESSION_SUB_TYPE_UNKNOWN | 0 | Session subtype unknown |
| SESSION_SUB_TYPE_SINGLE_HOP | 1 | Single-hop session |
| SESSION_SUB_TYPE_MULTI_HOP | 2 | Multi-hop session |
| SESSION_SUB_TYPE_MICROBFD | 3 | microbfd session |
| SESSION_SUB_TYPE_SBFDECHO | 4 | microbfd session |

4.2.2 BfdmgrGeneralSessionKeyPb.SbfdechoKey.SrpolicyUserType

Table 6: *BfdmgrGeneralSessionKeyPb.SbfdechoKey.SrpolicyUserType*

| Name | Number | Description |
|------------|--------|-------------|
| TYPE_LOCAL | 0 | |
| TYPE_PCEP | 1 | |

4.2.3 BfdmgrSessionStatus

Represents BFD session status.

Table 7: *BfdmgrSessionStatus*

| Name | Number | Description |
|------------|--------|-----------------------|
| INVALID | 0 | Session invalid |
| ADMIN_DOWN | 1 | Admin down |
| DOWN | 2 | Status down |
| INIT | 3 | Status initializing |
| UP | 4 | Status up and running |

4.2.4 BfdmgrSessionType

Represents BFD session type.

Table 8: *BfdmgrSessionType*

| Name | Number | Description |
|-------------------------|--------|---|
| SESSION_TYPE_UNKNOWN | 0 | Unknown session type |
| SESSION_TYPE_P2P | 1 | Peer-to-peer session type |
| SESSION_TYPE_MICROBFD | 2 | microbfd session type |
| SESSION_TYPE_SBFID_ECHO | 3 | seamless BFD session type, echo initiator |

4.2.5 BfdSessionNotification

Represents BFD session notification.

Table 9: BfdSessionNotification

| Field | Type | Label | Description |
|-------|--|-------|---|
| op | SdkMgrOperation | | Operation such as session create, delete, or update |
| key | BfdmgrGeneralSessionKeyPb | | Session key |
| data | BfdmgrGeneralSessionDataPb | | Session data |

4.2.6 BfdSessionSubscriptionRequest

Represents BFD session subscription request.

Table 10: BfdSessionSubscriptionRequest

| Field | Type | Label | Description |
|-------|---|-------|-----------------------------|
| key | BfdmgrGeneralSessionKeyPb | | Optional, to filter on name |

4.2.7 BfdmgrGeneralSessionDataPb

Represents BFD session data.

Table 11: BfdmgrGeneralSessionDataPb

| Field | Type | Label | Description |
|-----------|--------------------------------------|-------|--|
| status | BfdmgrSessionStatus | | Status of the session |
| sub_type | BfdmgrSessionSubType | | Subtype of the session |
| src_if_id | uint32 | | src_if_id is only populated for P2P type Source interface ID |

4.2.8 BfdmgrGeneralSessionKeyPb

Represents BFD session key.

Table 12: BfdmgrGeneralSessionKeyPb

| Field | Type | Label | Description |
|-------|-----------------------------------|-------|---|
| type | BfdmgrSessionType | | type is always present, other key field presence is determined by type Session type |

| Field | Type | Label | Description |
|----------|-------------|-------|-------------|
| p2p | P2pKey | | |
| microbfd | MicrobfdKey | | |
| sbfdecho | SbfdechoKey | | |

4.2.9 BfdmgrGeneralSessionKeyPb.MicrobfdKey

Table 13: BfdmgrGeneralSessionKeyPb.MicrobfdKey

| Field | Type | Label | Description |
|----------------|--------|-------|-------------|
| interface_name | string | | |

4.2.10 BfdmgrGeneralSessionKeyPb.P2pKey

Table 14: BfdmgrGeneralSessionKeyPb.P2pKey

| Field | Type | Label | Description |
|-----------------------------|-------------|-------|--|
| src_ip_addr | IpAddressPb | | Source IP address of the session |
| dst_ip_addr | IpAddressPb | | Destination IP address of the session |
| instance_id | uint32 | | Network instance identifier |
| ipv4_unnum_or_ipv6_ll_if_id | uint32 | | Global if id for ipv4 unnumbered or ipv6 link local session, otherwise 0 |
| specified_discr | bool | | True if discriminators are specified |

4.2.11 BfdmgrGeneralSessionKeyPb.SbfdechoKey

Table 15: BfdmgrGeneralSessionKeyPb.SbfdechoKey

| Field | Type | Label | Description |
|---------------------------|--------|-------|-----------------------------|
| instance_id | uint32 | | Network instance identifier |
| sr_policy_segment_list_id | uint32 | | |
| policy_name | string | | |

| Field | Type | Label | Description |
|--------------------|----------------------------------|-------|-------------|
| user_type | SrpolicyUserType | | |
| sr_policy_endpoint | IpAddressPb | | |

4.3 config_service.proto

4.3.1 AcknowledgeConfigRequest

Represents configuration acknowledgment request; Each config notification requires its own acknowledgement.

Table 16: AcknowledgeConfigRequest

| Field | Type | Label | Description |
|-------|--|----------|-------------|
| infos | AcknowledgeConfigRequestInfo | repeated | |

4.3.2 AcknowledgeConfigRequestInfo

Represents configuration acknowledgment request information.

Table 17: AcknowledgeConfigRequestInfo

| Field | Type | Label | Description |
|-------------------|------------------------|-------|---|
| js_path_with_keys | string | | JSON path formatted string from YANG; for example, interface{.name==ethernet1/1}.my_field |
| error | string | | |
| warning | string | | |
| output | string | | |

4.3.3 AcknowledgeConfigResponse

Represents configuration acknowledgement response.

Table 18: AcknowledgeConfigResponse

| Field | Type | Label | Description |
|-----------|------------------------------|-------|--|
| status | SdkMgrStatus | | Status of acknowledge config request operation |
| error_str | string | | Detailed error string |

4.3.4 ConfigData

Represents configuration data.

Table 19: ConfigData

| Field | Type | Label | Description |
|-------|------------------------|-------|--|
| json | string | | Entire configuration fragment as JSON string |
| data | bytes | | Entire configuration fragment as binary data |

4.3.5 ConfigKey

Represents configuration key.

Table 20: ConfigKey

| Field | Type | Label | Description |
|-------------------|------------------------|----------|---|
| js_path | string | | JSON path formatted string from YANG; for example, interface.my_field |
| keys | string | repeated | Value for keys |
| js_path_with_keys | string | | JSON path formatted string from YANG; for example, interface{.name==ethernet1/1}.my_field |

4.3.6 ConfigNotification

Represents configuration notification message to subscribe to configuration events

Table 21: ConfigNotification

| Field | Type | Label | Description |
|-------|---------------------------------|-------|--|
| op | SdkMgrOperation | | Operation indicating create, delete, or update |
| key | ConfigKey | | Configuration key |
| data | ConfigData | | Configuration data |

4.3.7 ConfigSubscriptionRequest

Represents configuration subscription request.

Table 22: ConfigSubscriptionRequest

| Field | Type | Label | Description |
|-------|---------------------------|-------|-----------------------------|
| key | ConfigKey | | Optional, to filter on name |

4.3.8 SdkMgrConfigService

Represents service for config operations.

Table 23: SdkMgrConfigService

| Method Name | Request Type | Response Type | Description |
|-------------------|--|---|---|
| AcknowledgeConfig | AcknowledgeConfigRequest | AcknowledgeConfigResponse | Acknowledge received configuration. When agent is registered with `wait_config_ack` flag set to true, it needs to acknowledge received configuration. |

4.4 interface_service.proto

4.4.1 InterfaceData

Represents interface data.

Table 24: InterfaceData

| Field | Type | Label | Description |
|--------------|--------------|-------|---|
| admin_is_up | uint32 | | Admin state |
| mtu | uint32 | | Maximum transmission unit |
| if_type | IfMgrIfType | | Interface type; for example, loopback, physical, or LAG |
| port_id | PortIdPb | | Port identifier |
| description | string | | Interface description |
| mac_addr | MacAddressPb | | MAC address |
| aggregate_id | string | | associated aggregate id |
| oper_is_up | uint32 | | Operational state |

4.4.2 InterfaceKey

Represents interface key.

Table 25: InterfaceKey

| Field | Type | Label | Description |
|---------|--------|-------|---|
| if_name | string | | Interface name; for example, ethernet 1/1 |

4.4.3 InterfaceNotification

Represents interface notification.

Table 26: InterfaceNotification

| Field | Type | Label | Description |
|-------|-----------------|-------|---|
| op | SdkMgrOperation | | Operation such as create, delete, or update |
| key | InterfaceKey | | Interface key |
| data | InterfaceData | | Interface data |

4.4.4 InterfaceSubscriptionRequest

Represents interface subscription request.

Table 27: InterfaceSubscriptionRequest

| Field | Type | Label | Description |
|-------|------------------------------|-------|-----------------------------|
| key | InterfaceKey | | Optional, to filter on name |

4.5 lldp_service.proto

4.5.1 LldpNeighborDataPb.PortSubType

Represents port subtype.

Table 28: LldpNeighborDataPb.PortSubType

| Name | Number | Description |
|------------------|--------|--|
| RESERVED | 0 | Reserved for future use |
| INTERFACE_ALIAS | 1 | Alias of the interface |
| PORT_COMPONENT | 2 | Port identifier based on a locally defined port component |
| MAC_ADDRESS | 3 | MAC address |
| NETWORK_ADDRESS | 4 | Network address |
| INTERFACE_NAME | 5 | Name of the interface |
| AGENT_CIRCUIT_ID | 6 | Port identifier based on the circuit ID in the DHCP relay agent information option |
| LOCALLY_ASSIGNED | 7 | Port identifier based on a locally defined alphanumeric string |

4.5.2 LldpNeighborKeyPb.ChassisIdType

Represents chassis type.

Table 29: *LdpNeighborKeyPb.ChassisIdType*

| Name | Number | Description |
|-------------------|--------|---|
| RESERVED | 0 | Reserved for future use |
| CHASSIS_COMPONENT | 1 | Chassis identifier based on a locally defined chassis component |
| INTERFACE_ALIAS | 2 | Alias of the interface |
| PORT_COMPONENT | 3 | Chassis identifier based on a locally defined port component |
| MAC_ADDRESS | 4 | MAC address |
| NETWORK_ADDRESS | 5 | Network address |
| INTERFACE_NAME | 6 | Name of the interface |
| LOCALLY_ASSIGNED | 7 | Chassis identifier based on a locally defined value |

4.5.3 LdpNeighborDataPb

Represents LLDP neighbor data.

Table 30: *LdpNeighborDataPb*

| Field | Type | Label | Description |
|--------------------|------------------------------|----------|-----------------------------------|
| port_id | string | | Port identifier |
| port_type | PortSubType | | Port type |
| source_mac | MacAddressPb | | Port MAC address |
| bgp_peer_address | IpAddressPb | repeated | LLDP BGP autodiscovered addresses |
| bgp_group_id | uint32 | | BGP group identifier |
| system_name | string | | System name |
| system_description | string | | System description |

4.5.4 LdpNeighborKeyPb

Represents LLDP neighbor key.

Table 31: *LdpNeighborKeyPb*

| Field | Type | Label | Description |
|----------------|-------------------------------|-------|----------------------|
| interface_name | string | | Local interface name |
| chassis_id | string | | Chassis identifier |
| chassis_type | ChassisIdType | | Chassis type |

4.5.5 LdpNeighborNotification

Represents LLDP neighbor notification.

Table 32: *LdpNeighborNotification*

| Field | Type | Label | Description |
|-------|-----------------------------------|-------|---|
| op | SdkMgrOperation | | Operation such as create, delete, or update |
| key | LdpNeighborKeyPb | | LLDP neighbor key |
| data | LdpNeighborDataPb | | LLDP neighbor data |

4.5.6 LdpNeighborSubscriptionRequest

Represents LLDP neighbor subscription request.

Table 33: *LdpNeighborSubscriptionRequest*

| Field | Type | Label | Description |
|-------|----------------------------------|-------|-----------------------------|
| key | LdpNeighborKeyPb | | Optional, to filter on name |

4.6 networkinstance_service.proto

4.6.1 NetworkInstanceData.NetInstType

Represents network instance type.

Table 34: *NetworkInstanceData.NetInstType*

| Name | Number | Description |
|---------|--------|-------------------------------|
| DEFAULT | 0 | Default network instance type |

| Name | Number | Description |
|-------|--------|-----------------------------|
| L3VRF | 1 | L3VRF network instance type |

4.6.2 NetworkInstanceData

Represents network instance data.

Table 35: NetworkInstanceData

| Field | Type | Label | Description |
|-------------|-------------|-------|-----------------------------|
| net_inst_id | uint32 | | Network instance identifier |
| base_name | string | | Base name |
| oper_is_up | bool | | Operation status |
| router_id | string | | Router identifier |
| inst_type | NetInstType | | Network instance type |

4.6.3 NetworkInstanceKey

Represents network instance key.

Table 36: NetworkInstanceKey

| Field | Type | Label | Description |
|-----------|--------|-------|-----------------------|
| inst_name | string | | Network instance name |

4.6.4 NetworkInstanceNotification

Represents network instance notification.

Table 37: NetworkInstanceNotification

| Field | Type | Label | Description |
|-------|---------------------|-------|---|
| op | SdkMgrOperation | | Operation such as create, delete, or update |
| key | NetworkInstanceKey | | Network key |
| data | NetworkInstanceData | | Network data |

4.6.5 NetworkInstanceSubscriptionRequest

Represents network instance subscription request.

4.7 nexthop_group_service.proto

4.7.1 NextHop.ResolutionType

Represents resolution type.

Table 38: NextHop.ResolutionType

| Name | Number | Description |
|---------|--------|--------------------|
| INVALID | 0 | Invalid resolution |
| REGULAR | 1 | Regular resolution |
| MPLS | 2 | MPLS resolution |

4.7.2 NextHop.ResolveToType

Represents resolve-to type.

Table 39: NextHop.ResolveToType

| Name | Number | Description |
|----------|--------|----------------------------|
| LOCAL | 0 | Resolve to local routes |
| DIRECT | 1 | Resolve to direct routes |
| INDIRECT | 2 | Resolve to indirect routes |

4.7.3 MplsNextHop

Represents MPLS next hop.

Table 40: MplsNextHop

| Field | Type | Label | Description |
|-------------|-----------------------------|----------|---------------------|
| ip_nexthop | IpAddressPb | | Next-hop IP address |
| label_stack | MplsLabel | repeated | MPLS label stack |

4.7.4 NextHop

Represents next-hop.

Table 41: NextHop

| Field | Type | Label | Description |
|--------------|--------------------------------|-------|---------------------|
| resolve_to | ResolveToType | | Resolve-to type |
| type | ResolutionType | | Resolution type |
| ip_nexthop | IpAddressPb | | IP next-hop address |
| mpls_nexthop | MplsNextHop | | MPLS next-hop |

4.7.5 NextHopGroup

Represents next-hop group.

Table 42: NextHopGroup

| Field | Type | Label | Description |
|----------|-------------------------|----------|-------------|
| next_hop | NextHop | repeated | Next-hops |

4.7.6 NextHopGroupDeleteRequest

Represents next-hop group delete request.

Table 43: NextHopGroupDeleteRequest

| Field | Type | Label | Description |
|-----------|---------------------------------|----------|----------------------------|
| group_key | NextHopGroupKey | repeated | Next-hop group key details |

4.7.7 NextHopGroupDeleteResponse

Represents next-hop group delete response.

Table 44: NextHopGroupDeleteResponse

| Field | Type | Label | Description |
|-----------|------------------------------|-------|-------------------------------------|
| status | SdkMgrStatus | | Response for next-hop group request |
| error_str | string | | Detailed error string |

4.7.8 NextHopGroupInfo

Represents next-hop group information.

Table 45: NextHopGroupInfo

| Field | Type | Label | Description |
|-------|---------------------------------|-------|---------------------|
| key | NextHopGroupKey | | Next-hop group key |
| data | NextHopGroup | | Next-hop group data |

4.7.9 NextHopGroupKey

Represents next-hop group key.

Table 46: NextHopGroupKey

| Field | Type | Label | Description |
|-----------------------|------------------------|-------|--------------------------------------|
| name | string | | Next-hop group name |
| network_instance_name | string | | Next-hop group network instance name |

4.7.10 NextHopGroupNotification

Represents next-hop group notification.

Table 47: NextHopGroupNotification

| Field | Type | Label | Description |
|-------|---------------------------------|-------|---|
| op | SdkMgrOperation | | Operation such as create, delete, or update |
| key | uint64 | | Next-hop group key |
| data | NextHopGroup | | Next-hop group data |

4.7.11 NextHopGroupRequest

Represents next-hop group request.

Table 48: NextHopGroupRequest

| Field | Type | Label | Description |
|------------|----------------------------------|----------|------------------------|
| group_info | NextHopGroupInfo | repeated | Next-hop group details |

4.7.12 NextHopGroupResponse

Represents next-hop group response.

Table 49: NextHopGroupResponse

| Field | Type | Label | Description |
|-----------|------------------------------|-------|-------------------------------------|
| status | SdkMgrStatus | | Response for next-hop group request |
| error_str | string | | Detailed error string |

4.7.13 NextHopGroupSubscriptionRequest

Represents next-hop group subscription request.

Table 50: NextHopGroupSubscriptionRequest

| Field | Type | Label | Description |
|-------|---------------------------------|-------|-----------------------------|
| key | NextHopGroupKey | | Optional, to filter on name |

4.7.14 SdkMgrNextHopGroupService

Represents service for next-hop group operations.

Table 51: SdkMgrNextHopGroupService

| Method Name | Request Type | Response Type | Description |
|--------------------------|---|---|--|
| NextHopGroupAdd OrUpdate | NextHopGroupRequest | NextHopGroup Response | Add or update one or more next-hop groups. |
| NextHopGroupDelete | NextHopGroupDeleteRequest | NextHopGroup DeleteResponse | Delete next-hop group. |
| SyncStart | SyncRequest | SyncResponse | Synchronization start to open synchronization operation. |
| SyncEnd | SyncRequest | SyncResponse | Synchronization end to close synchronization operation. |

4.8 route_service.proto

4.8.1 IpRouteNotification

Represents IP route notification.

Table 52: IpRouteNotification

| Field | Type | Label | Description |
|-------|---------------------------------|-------|---|
| op | SdkMgrOperation | | Operation such as create, delete, or update |
| key | RouteKeyPb | | IP route key |
| data | RoutePb | | IP route data |

4.8.2 IpRouteSubscriptionRequest

Represents IP route subscription request.

Table 53: IpRouteSubscriptionRequest

| Field | Type | Label | Description |
|-------|----------------------------|-------|-----------------------------|
| key | RouteKeyPb | | Optional, to filter on name |

4.8.3 RouteAddRequest

Represents route add request; can contain more than one route.

Table 54: RouteAddRequest

| Field | Type | Label | Description |
|--------|---------------------------|----------|-------------|
| routes | RouteInfo | repeated | IP routes |

4.8.4 RouteAddResponse

Represents route add response.

Table 55: RouteAddResponse

| Field | Type | Label | Description |
|-----------|------------------------------|-------|-------------------------------|
| status | SdkMgrStatus | | Status of route add operation |
| error_str | string | | Detailed error string |

4.8.5 RouteDeleteRequest

Represents route delete request; can contain more than one route.

Table 56: RouteDeleteRequest

| Field | Type | Label | Description |
|--------|----------------------------|----------|-------------|
| routes | RouteKeyPb | repeated | IP routes |

4.8.6 RouteDeleteResponse

Represents route delete response.

Table 57: RouteDeleteResponse

| Field | Type | Label | Description |
|-----------|------------------------------|-------|----------------------------------|
| status | SdkMgrStatus | | Status of route delete operation |
| error_str | string | | Detailed error string |

4.8.7 RouteInfo

Represents route information.

Table 58: RouteInfo

| Field | Type | Label | Description |
|-------|----------------------------|-------|-------------|
| key | RouteKeyPb | | Route key |
| data | RoutePb | | Route data |

4.8.8 RouteKeyPb

Represents route key.

Table 59: RouteKeyPb

| Field | Type | Label | Description |
|---------------|---------------------------------|-------|-----------------------|
| net_inst_name | string | | Network instance name |
| ip_prefix | IpAddrPrefLenPb | | IP prefix |

4.8.9 RoutePb

Represents route data.

Table 60: RoutePb

| Field | Type | Label | Description |
|--------------------|-------------------------|----------|--|
| nexthop_group_name | string | | Next hop group name |
| preference | uint32 | | Preference |
| metric | uint32 | | Metric |
| nexthop | NextHop | repeated | List of next hops |
| owner_id | uint32 | | Next hop owner identifier returned only on notification. |
| nhg_id | uint64 | | Next-hop group identifier returned only on notification. |

4.8.10 SdkMgrRouteService

Represents service for IP route operations.

Table 61: SdkMgrRouteService

| Method Name | Request Type | Response Type | Description |
|------------------|------------------------------------|-------------------------------------|-------------------------------------|
| RouteAddOrUpdate | RouteAddRequest | RouteAddResponse | Add or update IP routes. |
| RouteDelete | RouteDeleteRequest | RouteDeleteResponse | Delete IP routes. |
| SyncStart | SyncRequest | SyncResponse | Synchronization start for IP routes |
| SyncEnd | SyncRequest | SyncResponse | Synchronization end for IP routes |

4.9 sdk_common.proto

4.9.1 IfEthernetDuplexModeType

Represents interface ethernet duplex mode. Corresponds to yang values

Table 62: IfEthernetDuplexModeType

| Name | Number | Description |
|--------------------------|--------|---------------------------|
| IF_ETH_DUPLEX_MODE_UNSET | 0 | duplex mode not supported |
| IF_ETH_DUPLEX_MODE_FULL | 1 | |
| IF_ETH_DUPLEX_MODE_HALF | 2 | |

4.9.2 IfEthernetPortSpeedType

Represents interface ethernet port speed. Corresponds to yang values

Table 63: IfEthernetPortSpeedType

| Name | Number | Description |
|-------------------------|--------|---------------|
| IF_ETH_PORT_SPEED_UNSET | 0 | Speed unknown |
| IF_ETH_PORT_SPEED_10M | 1 | |
| IF_ETH_PORT_SPEED_100M | 2 | |
| IF_ETH_PORT_SPEED_1G | 3 | |
| IF_ETH_PORT_SPEED_10G | 4 | |
| IF_ETH_PORT_SPEED_25G | 5 | |
| IF_ETH_PORT_SPEED_40G | 6 | |
| IF_ETH_PORT_SPEED_50G | 7 | |
| IF_ETH_PORT_SPEED_100G | 8 | |
| IF_ETH_PORT_SPEED_200G | 9 | |

| Name | Number | Description |
|------------------------|--------|-------------|
| IF_ETH_PORT_SPEED_400G | 10 | |
| IF_ETH_PORT_SPEED_1T | 11 | |
| IF_ETH_PORT_SPEED_800G | 12 | |

4.9.3 IfLoopbackModeType

Represents interface loopback mode. Corresponds to yang values

Table 64: IfLoopbackModeType

| Name | Number | Description |
|---------------------------|--------|-----------------------------|
| IF_LOOPBACK_MODE_UNSET | 0 | loopback mode not supported |
| IF_LOOPBACK_MODE_NONE | 1 | |
| IF_LOOPBACK_MODE_FACILITY | 2 | |
| IF_LOOPBACK_MODE_TERMINAL | 3 | |

4.9.4 IfMgrIfType

Represents interface type.

Table 65: IfMgrIfType

| Name | Number | Description |
|------------|--------|---|
| ETHERNET | 0 | Ethernet interface |
| LOOPBACK | 1 | Loopback interface |
| MANAGEMENT | 2 | Management interface |
| AGGREGATE | 3 | Aggregate(LAG) interface |
| IRB | 4 | Integrated Routing and Bridging (IRB) interface |
| SYSTEM | 5 | System interface |

| Name | Number | Description |
|-------------|--------|--|
| LIF | 6 | linux interface |
| NIC | 7 | linux nic interface (bus/dev/fn) |
| VHOST | 8 | vhost-net interface, vhn-<name> name for sock-path |
| KKLIF | 9 | temp name for new style of lif interface |
| KKVHOST | 10 | temp name for new style of vhost interface |
| SYNC | 11 | 1588 sync interface |
| IF_TYPE_MAX | 12 | |

4.9.5 IfOperDownReason

Table 66: IfOperDownReason

| Name | Number | Description |
|-----------------------------------|--------|-------------|
| IF_OPER_DOWN_NONE | 0 | |
| IF_OPER_DOWN_PORT_ADMIN_DISABLED | 1 | |
| IF_OPER_DOWN_MDA_ADMIN_DISABLED | 2 | |
| IF_OPER_DOWN_TRANS_LASER_DISABLED | 3 | |
| IF_OPER_DOWN_MDA_NOT_PRESENT | 4 | |
| IF_OPER_DOWN_TRANS_NOT_PRESENT | 5 | |
| IF_OPER_DOWN_PHY_INIT | 6 | |
| IF_OPER_DOWN_LOWER_LAYER_DOWN | 7 | |
| IF_OPER_DOWN_MTU_RESOURCES | 8 | |
| IF_OPER_DOWN_UNSUPPORTED_SPEED | 9 | |

| Name | Number | Description |
|------------------------------------|--------|---|
| IF_OPER_DOWN_UNSUPPORTED_TRANS_FEC | 10 | |
| IF_OPER_DOWN_OTHER | 11 | |
| IF_OPER_DOWN_PORT_NOT_PRESENT | 12 | used internally by chassis mgr only - xdp never publish to IDB! |
| IF_OPER_DOWN_FABRIC_AVAILABILITY | 13 | used internally by chassis mgr only - xdp never publish to IDB! |
| IF_OPER_DOWN_NO_ACTIVE_LINKS | 14 | lag interface only |
| IF_OPER_DOWN_MIN_LINK_THRESHOLD | 15 | lag interface only |
| IF_OPER_DOWN_9_12_SPEED_MISMATCH | 16 | Vodka port 9-12 must all be same speed as port 9 |
| IF_OPER_DOWN_LAG_RESOURCES | 17 | lag interface only |
| IF_OPER_DOWN_LAG_MEMBER_RESOURCES | 18 | lag member interface only |
| IF_OPER_DOWN_STANDBY_SIGNALING | 19 | ESM multihoming |
| IF_OPER_DOWN_HOLD_TIME_UP_ACTIVE | 20 | interface hold-time up is actively holding the interface down |
| IF_OPER_DOWN_RELOAD_TIME_ACTIVE | 21 | interface reload time is actively holding the interface down |
| IF_OPER_DOWN_CONNECTOR_DOWN | 22 | parent connector oper down forces breakout port oper down |
| IF_OPER_DOWN_AUTO_NEG_MISMATCH | 23 | |
| IF_OPER_DOWN_EVENT_HANDLER | 24 | used internally by chassis mgr only - xdp never publish to IDB! |
| IF_OPER_DOWN_UNSUPPORTED_BREAKOUT | 25 | interface doesn't support breakout config |
| IF_OPER_DOWN_CFM_CCM_DEFECT | 26 | |

| Name | Number | Description |
|---|--------|--|
| IF_OPER_DOWN_CRC_MON_FAIL_THRESH | 27 | crc-monitor signal failure threshold exceeded |
| IF_OPER_DOWN_SYMBOL_MON_FAIL_THRESH | 28 | symbol-monitor signal failure threshold exceeded |
| IF_OPER_DOWN_LINK_LOSS_FORWARDING | 29 | related to evpn-vpws mpls |
| IF_OPER_DOWN_TRANS_DOWN | 30 | |
| IF_OPER_DOWN_STORM_CONTROL_ACTION | 31 | |
| IF_OPER_DOWN_UNSUPPORTED_NUM_CHANNELS_FOR_SPEED | 32 | |

4.9.6 IfOperStateType

Represents interface operational state.

Table 67: IfOperStateType

| Name | Number | Description |
|--------------------------------|--------|--|
| IF_OPER_STATE_UP | 0 | Interface operational state up |
| IF_OPER_STATE_DOWN | 1 | Interface operational state down |
| IF_OPER_STATE_TESTING | 2 | Interface operational state testing |
| IF_OPER_STATE_UNKNOWN | 3 | Interface operational state unknown |
| IF_OPER_STATE_DORMANT | 4 | Interface operational state dormant |
| IF_OPER_STATE_NOT_PRESENT | 5 | Interface operational state not present |
| IF_OPER_STATE_LOWER_LAYER_DOWN | 6 | Interface operational state lower layer down |

4.9.7 IfTransceiverFecType

Represents interface transceiver fec. Corresponds to yang values

Table 68: *IfTransceiverFecType*

| Name | Number | Description |
|-----------------------|--------|-------------|
| IF_TRANS_FEC_UNSET | 0 | Fec unknown |
| IF_TRANS_FEC_DISABLED | 1 | |
| IF_TRANS_FEC_RS528 | 2 | |
| IF_TRANS_FEC_RS544 | 3 | |
| IF_TRANS_FEC_BASER | 4 | |
| IF_TRANS_FEC_RS108 | 5 | |

4.9.8 IpAddressState

Represents IP address state.

Table 69: *IpAddressState*

| Name | Number | Description |
|---------------------------|--------|-------------------------------|
| IPADDR_STATE_UNKNOWN | 0 | IP address state unknown |
| IPADDR_STATE_TENTATIVE | 1 | IP address state tentative |
| IPADDR_STATE_DUPPLICATED | 2 | IP address state duplicated |
| IPADDR_STATE_INACCESSIBLE | 3 | IP address state inaccessible |
| IPADDR_STATE_DEPRECATED | 4 | IP address state deprecated |
| IPADDR_STATE_PREFERRED | 5 | IP address state preferred |

4.9.9 PortOperDownReason

Various reasons for port being operationally down.

Table 70: *PortOperDownReason*

| Name | Number | Description |
|---------------------|--------|-------------|
| PORT_OPER_DOWN_NONE | 0 | |

| Name | Number | Description |
|--|--------|--|
| PORT_OPER_DOWN_PORT_ADMIN_DISABLED | 1 | Admin disabled (out of service). E.g lanes not created in physical layer devices |
| PORT_OPER_DOWN_PORT_OPER_DISABLED | 2 | Admin enabled but operationally disabled. E.g lanes created but serdes are off |
| PORT_OPER_DOWN_MDA_DISABLED | 3 | Parent MDA is disabled |
| PORT_OPER_DOWN_MDA_FAILED | 4 | Parent MDA failed |
| PORT_OPER_DOWN_LINK_DOWN | 5 | Operationally enabled but link/carrier down |
| PORT_OPER_DOWN_TRANS_NOT_PRESENT | 6 | Transceiver is not plugged in or is dead (undetected) |
| PORT_OPER_DOWN_TRANS_LASER_DISABLED | 7 | Transceiver laser is disabled |
| PORT_OPER_DOWN_INCOMPATIBLE_CONNECTOR | 8 | Parent connector configuration does not match with this breakout port |
| PORT_OPER_DOWN_PHY_UNSUPPORTED_FEC_CFG | 9 | A physical layer device on the port does not support configured FEC |
| PORT_OPER_DOWN_PHY_UNSUPPORTED_TRANS_FEC | 10 | A physical layer device on the port does not support FEC configured on the transceiver |
| PORT_OPER_DOWN_PHY_INTERNAL_FEC_ISSUE | 11 | A physical layer device has internal/hardware problem on employing FEC |
| PORT_OPER_DOWN_UNSUPPORTED_BREAKOUT | 12 | Connector does not support the configured breakout mode. |

4.9.10 PortWarningReason

Various warnings on port while it stays operational.

Table 71: PortWarningReason

| Name | Number | Description |
|-------------------|--------|-------------|
| PORT_WARNING_NONE | 0 | |

| Name | Number | Description |
|---|--------|--|
| PORT_WARNING_TRANS_RX_LOS | 1 | Transceiver reported RxLos |
| PORT_WARNING_TRANS_UNSUPPORTED_SYNC_CFG | 2 | Transceiver does not support synce configuration (e.g cannot do clock squelching on RxLos) |
| PORT_WARNING_PHY_UNSUPPORTED_SYNC_CFG | 3 | A physical layer device does not support synce configuration |
| PORT_WARNING_PHY_UNSUPPORTED_PTP_CFG | 4 | A physical layer device does not support PTP configuration |

4.9.11 SdkMgrOperation

Represents enumeration value for operation in subscription.

Table 72: SdkMgrOperation

| Name | Number | Description |
|----------------|--------|---|
| Create | 0 | Create operation; returned if caching is enabled |
| Update | 1 | Update operation; returned if caching is enabled |
| Delete | 2 | Delete operation |
| CreateOrUpdate | 3 | returned if caching is disabled; App can cache streaming data, if needed. |

4.9.12 SdkMgrStatus

Represents status of network programming service calls.

Table 73: SdkMgrStatus

| Name | Number | Description |
|----------------|--------|-------------------------|
| kSdkMgrSuccess | 0 | Successful service call |
| kSdkMgrFailed | 1 | Failed service call |

4.9.13 AgentReply

Empty message from agent.

4.9.14 EvpnEthSegIdPb

Table 74: EvpnEthSegIdPb

| Field | Type | Label | Description |
|-------|-------|-------|-------------------------------|
| es_id | bytes | | Type 0 for now. hard-coded id |

4.9.15 GlobalIfId

Represents global interface identifier.

Table 75: GlobalIfId

| Field | Type | Label | Description |
|--------------|--------|-------|-----------------------------|
| global_if_id | uint32 | | Global interface identifier |

4.9.16 IpAddrPrefLenPb

Represents IP prefix.

Table 76: IpAddrPrefLenPb

| Field | Type | Label | Description |
|---------------|-------------|-------|--------------------------|
| ip_addr | IpAddressPb | | IP address |
| prefix_length | uint32 | | IP address prefix length |

4.9.17 IpAddressPb

Represents IP address.

Table 77: IpAddressPb

| Field | Type | Label | Description |
|-------|-------|-------|-------------|
| addr | bytes | | IP address |

4.9.18 IpInterfaceAddrPrefixPb

Represents IP prefix state.

Table 78: IpInterfaceAddrPrefixPb

| Field | Type | Label | Description |
|--------|---------------------------------|-------|-----------------|
| prefix | IpAddrPrefLenPb | | IP prefix |
| state | IpAddressState | | IP prefix state |

4.9.19 MacAddressPb

Represents MAC address.

Table 79: MacAddressPb

| Field | Type | Label | Description |
|-------------|-----------------------|-------|-------------|
| mac_address | bytes | | MAC address |

4.9.20 MplsLabel

Represents MPLS label.

Table 80: MplsLabel

| Field | Type | Label | Description |
|------------|------------------------|-------|-------------|
| mpls_label | uint32 | | MPLS label |

4.9.21 NetInstanceld

Represents network instance identifier.

Table 81: NetInstanceld

| Field | Type | Label | Description |
|-------------|------------------------|-------|-----------------------------|
| instance_id | uint32 | | Network instance identifier |

4.9.22 PortIdPb

Represents port identifier.

Table 82: PortIdPb

| Field | Type | Label | Description |
|---------|--------|-------|-----------------|
| port_id | uint64 | | Port identifier |

4.9.23 SyncRequest

Empty message for synchronization request.

4.9.24 SyncResponse

Empty message for synchronization end.

Table 83: SyncResponse

| Field | Type | Label | Description |
|-----------|--------------|-------|-----------------------|
| status | SdkMgrStatus | | Error code |
| error_str | string | | Detailed error string |

4.10 sdk_service.proto

4.10.1 NotificationRegisterRequest.Operation

Represents notification stream subscription request operation.

Table 84: NotificationRegisterRequest.Operation

| Name | Number | Description |
|--------------------|--------|---|
| Create | 0 | Create a subscription |
| Delete | 1 | Delete all subscriptions |
| AddSubscription | 2 | Add subscription to existing subscriptions |
| DeleteSubscription | 3 | Delete one subscription from existing subscriptions |

4.10.2 AgentRegistrationRequest

Represents registration request message used in agent register and unregister.

Table 85: AgentRegistrationRequest

| Field | Type | Label | Description |
|----------------------|--------|----------|--|
| js_path | string | repeated | Optional, JSON path formatted strings, which are used in telemetry. Format of js_path follows hierarchical YANG. for example: .interface{.name=*}.my_app. .my_app.tunnel{.name==*} "*" needs to be replaced with a specific key. |
| agent_liveliness | uint32 | | Kill this agent unless a keepalive is received within this many seconds. Value of 0 means do not monitor this agent for liveliness. |
| wait_config_ack | bool | | Indicate if SRLinux should wait for explicit ack from app after delivering configuration. |
| enable_cache | bool | | Indicate if SRLinux should cache streaming notification response in ndk mgr. By default, caching is disabled. |
| auto_telemetry_state | bool | | Indicate if SRLinux should automatically push configs as state. By default, auto_telemetry_state is disabled. |

4.10.3 AgentRegistrationResponse

Represents registration response in reply to registration request.

Table 86: AgentRegistrationResponse

| Field | Type | Label | Description |
|-----------|--------------|-------|---|
| status | SdkMgrStatus | | Status of the register; for example: kOk, kFailed |
| error_str | string | | Detailed error text |
| app_id | uint32 | | Application ID assigned by SDK manager. |

4.10.4 AppldRequest

Represents application identifier request from agent. All applications are assigned an identifier by IDB.

Table 87: AppldRequest

| Field | Type | Label | Description |
|-------|--------|-------|------------------|
| name | string | | Application name |

4.10.5 AppldResponse

Represents application identifier response to agent.

Table 88: AppldResponse

| Field | Type | Label | Description |
|--------|--------------|-------|---|
| status | SdkMgrStatus | | Status of the call; for example, kOk, kFailed |
| id | uint32 | | Identifier for the given application name |

4.10.6 KeepAliveRequest

Represents keep alive request from agent to refresh liveliness of the agent.

4.10.7 KeepAliveResponse

Represents keepalive response.

Table 89: KeepAliveResponse

| Field | Type | Label | Description |
|--------|--------------|-------|--|
| status | SdkMgrStatus | | Status of keepalive; for example, kOk or kFailed |

4.10.8 Notification

Represents notification stream response.

Table 90: Notification

| Field | Type | Label | Description |
|---------------|---|-------|----------------------------|
| sub_id | uint64 | | Subscription identifier |
| intf | InterfaceNotification | | Interface details |
| nw_inst | NetworkInstanceNotification | | Network instance details |
| lldp_neighbor | LldpNeighborNotification | | LLDP neighbor details |
| config | ConfigNotification | | Configuration notification |
| bfd_session | BfdSessionNotification | | BFD session details |
| route | IpRouteNotification | | IP route details |
| appid | AppldentNotification | | App identification details |
| nhg | NextHopGroupNotification | | Next-hop group details |

4.10.9 NotificationQueryRequest

Represents notification query to return specific subscription details.

Table 91: NotificationQueryRequest

| Field | Type | Label | Description |
|-----------|------------------------|-------|---|
| stream_id | uint64 | | Stream identifier, in Notification RegisterResponse |

4.10.10 NotificationQueryResponse

Represents notification query response.

Table 92: NotificationQueryResponse

| Field | Type | Label | Description |
|---------------|---|----------|------------------------------|
| subscriptions | NotificationQuerySubscription | repeated | List of subscription details |
| status | SdkMgrStatus | | Status of the query |

4.10.11 NotificationQuerySubscription

Represents notification subscription.

Table 93: NotificationQuerySubscription

| Field | Type | Label | Description |
|-------------|--------|-------|--------------------------|
| sub_id | uint64 | | Subscription identifier |
| description | string | | Subscription description |

4.10.12 NotificationRegisterRequest

Represents notification request from agent. Agent uses this message to subscribe to router events such as interface create, delete, or update, as well as LLDP neighbor create, delete, or update, and so on.

Table 94: NotificationRegisterRequest

| Field | Type | Label | Description |
|---------------|------------------------------------|-------|---|
| stream_id | uint64 | | Unset on create, set otherwise |
| op | Operation | | Specific operation in the notification register request |
| sub_id | uint64 | | Set for delete subscription, unset otherwise |
| intf | InterfaceSubscriptionRequest | | Interface subscription request |
| nw_inst | NetworkInstanceSubscriptionRequest | | Network instance subscription request |
| lldp_neighbor | LldpNeighborSubscriptionRequest | | LLDP neighbor subscription request |
| config | ConfigSubscriptionRequest | | Configuration subscription request |
| bfd_session | BfdSessionSubscriptionRequest | | BFD session subscription request |
| route | IpRouteSubscriptionRequest | | IP route subscription request |
| appid | AppIdentSubscriptionRequest | | App identification subscription request |
| nhg | NextHopGroupSubscriptionRequest | | Nexthop Group subscription request |

4.10.13 NotificationRegisterResponse

Represents notification response.

Table 95: NotificationRegisterResponse

| Field | Type | Label | Description |
|-----------|--------------|-------|--|
| stream_id | uint64 | | Stream identifier. This needs to be passed to the SDK manager for further notification subscription changes specific to the current subscription |
| sub_id | uint64 | | Subscription identifier. Each subscription gets an identifier, which can be used to delete a subscription |
| status | SdkMgrStatus | | Status of subscription |

4.10.14 NotificationStreamRequest

Represents notification stream request.

Table 96: NotificationStreamRequest

| Field | Type | Label | Description |
|-----------|--------|-------|-------------------|
| stream_id | uint64 | | Stream identifier |

4.10.15 NotificationStreamResponse

Represents notification stream response that contains one or more notification.

Table 97: NotificationStreamResponse

| Field | Type | Label | Description |
|--------------|--------------|----------|----------------------|
| notification | Notification | repeated | Notification details |

4.10.16 SdkMgrService

Represents base service that defines agent registration, unregistration, notification subscriptions, and keepalive messages.

Table 98: SdkMgrService

| Method Name | Request Type | Response Type | Description |
|----------------------|---|--|---|
| AgentRegister | AgentRegistrationRequest | AgentRegistrationResponse | Register agent |
| AgentUnRegister | AgentRegistrationRequest | AgentRegistrationResponse | Unregister agent |
| NotificationRegister | NotificationRegisterRequest | NotificationRegisterResponse | Register for event notifications |
| NotificationQuery | NotificationQueryRequest | NotificationQueryResponse | Returns current or specific notification subscription details |
| KeepAlive | KeepAliveRequest | KeepAliveResponse | Send periodic keepalive message |
| GetAppld | AppldRequest | AppldResponse | Get application name from application identifier |

4.10.17 SdkNotificationService

Represents service for handling notifications.

Table 99: SdkNotificationService

| Method Name | Request Type | Response Type | Description |
|--------------------|---|--|---|
| NotificationStream | NotificationStreamRequest | NotificationStreamResponse | Send stream of event notifications based on the agent subscriptions |

4.11 telemetry_service.proto

4.11.1 TelemetryData

Represents telemetry data.

Table 100: TelemetryData

| Field | Type | Label | Description |
|--------------|------------------------|-------|--------------------------------|
| json_content | string | | Structured JSON telemetry data |

4.11.2 TelemetryDeleteRequest

Represents telemetry delete request.

Table 101: TelemetryDeleteRequest

| Field | Type | Label | Description |
|-------|------------------------------|----------|---------------|
| key | TelemetryKey | repeated | Telemetry key |

4.11.3 TelemetryDeleteResponse

Represents telemetry delete response.

Table 102: TelemetryDeleteResponse

| Field | Type | Label | Description |
|-----------|------------------------------|-------|--------------------------|
| status | SdkMgrStatus | | Status of delete request |
| error_str | string | | Detailed error message |

4.11.4 TelemetryInfo

Represents telemetry information.

Table 103: TelemetryInfo

| Field | Type | Label | Description |
|-------|-------------------------------|-------|----------------|
| key | TelemetryKey | | Telemetry key |
| data | TelemetryData | | Telemetry data |

4.11.5 TelemetryKey

Represents telemetry key.

Table 104: TelemetryKey

| Field | Type | Label | Description |
|---------|------------------------|-------|--|
| js_path | string | | JSON path referencing the key for telemetry data |

4.11.6 TelemetryUpdateRequest

Represents telemetry update request.

Table 105: TelemetryUpdateRequest

| Field | Type | Label | Description |
|-------|-------------------------------|----------|----------------------|
| state | TelemetryInfo | repeated | State of application |

4.11.7 TelemetryUpdateResponse

Represents telemetry update response.

Table 106: TelemetryUpdateResponse

| Field | Type | Label | Description |
|-----------|------------------------------|-------|------------------------------------|
| status | SdkMgrStatus | | Status of telemetry update request |
| error_str | string | | Detailed error message |

4.11.8 SdkMgrTelemetryService

Represents service for telemetry service to store state data.

Table 107: SdkMgrTelemetryService

| Method Name | Request Type | Response Type | Description |
|-----------------------|--|--|------------------------------|
| TelemetryAdd OrUpdate | TelemetryUpdateRequest | TelemetryUpdate Response | Add or update telemetry data |
| TelemetryDelete | TelemetryDeleteRequest | TelemetryDelete Response | Delete telemetry data |

4.12 Scalar Value Types

4.12.1 Scalar

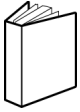
Scalar description

Table 108: Scalar

| .proto Type | Notes | C++ | C# | Go | Java | PHP | Python | Ruby |
|-------------|---|--------|--------|---------|--------|----------------|----------|--------------------------------|
| double | | double | double | float64 | double | float | float | Float |
| float | | float | float | float32 | float | float | float | Float |
| int32 | Uses variable-length encoding. Inefficient for encoding negative numbers – if your field is likely to have negative values, use sint32 instead. | int32 | int | int32 | int | integer | int | Bignum or Fixnum (as required) |
| int64 | Uses variable-length encoding. Inefficient for encoding negative numbers – if your field is likely to have negative values, use sint64 instead. | int64 | long | int64 | long | integer/string | int/long | Bignum |
| uint32 | Uses variable-length encoding. | uint32 | uint | uint32 | int | integer | int/long | Bignum or Fixnum (as required) |
| uint64 | Uses variable-length encoding. | uint64 | ulong | uint64 | long | integer/string | int/long | Bignum or Fixnum (as required) |
| sint32 | Uses variable-length encoding. Signed int value. These more efficiently encode negative numbers than regular int32s. | int32 | int | int32 | int | integer | int | Bignum or Fixnum (as required) |
| sint64 | Uses variable-length encoding. Signed int value. These more efficiently encode negative numbers than regular int64s. | int64 | long | int64 | long | integer/string | int/long | Bignum |

| .proto Type | Notes | C++ | C# | Go | Java | PHP | Python | Ruby |
|-------------|---|--------|-------------|--------|-------------|----------------|-------------|--------------------------------|
| fixed32 | Always four bytes. More efficient than uint32 if values are often greater than 2 ²⁸ . | uint32 | uint | uint32 | int | integer | int | Bignum or Fixnum (as required) |
| fixed64 | Always eight bytes. More efficient than uint64 if values are often greater than 2 ⁵⁶ . | uint64 | ulong | uint64 | long | integer/string | int/long | Bignum |
| sfixed32 | Always four bytes. | int32 | int | int32 | int | integer | int | Bignum or Fixnum (as required) |
| sfixed64 | Always eight bytes. | int64 | long | int64 | long | integer/string | int/long | Bignum |
| bool | | bool | bool | bool | boolean | boolean | boolean | TrueClass/False Class |
| string | A string must always contain UTF-8 encoded or 7-bit ASCII text. | string | string | string | String | string | str/unicode | String (UTF-8) |
| bytes | May contain any arbitrary sequence of bytes. | string | Byte String | []byte | Byte String | string | str | String (ASCII-8BIT) |

Customer document and product support



Customer documentation

[Customer documentation welcome page](#)



Technical support

[Product support portal](#)



Documentation feedback

[Customer documentation feedback](#)