Workgroup: Network Working Group
Internet-Draft:
draft-li-rtgwg-protocol-assisted-protocol-05
Published: 11 March 2023
Intended Status: Standards Track
Expires: 12 September 2023
Authors: Z. Li   S. Chen   Z. Tan   Y. Qu   Y. Gu
Huawei   Huawei   Huawei   Futurewei   Huawei

Protocol Assisted Protocol (PASP)

Abstract

For routing protocol troubleshooting, different approaches exibit merits w.r.t. different situations. They can be generally divided into two categories, the distributive way and the centralized way. A very commonly used distributive approach is to log in possiblly all related devices one by one to check massive data via CLI. Such approach provides very detailed device information, however it requires operators with high NOC (Network Operation Center) experience and suffers from low troubleshooting efficiency and high cost. The centralized approach is realized by collecting data from devices via approaches, like the streaming Telemetry or BMP(BGP Monitoring Protocol), for the centralized server to analyze all gathered data. Such approach allows a comprehensive view fo the whole network and facilitates automated troubleshooting, but is limited by the data collection boundary set by different management domains, as well as high network bandwidth and CPU computation costs.

This document proposes a semi-distributive and semi-centralized approach for fast routing protocol troubleshooting, localizing the target device and possibly the root cause, more precisely. It defines a new protocol, called the PASP (Protocol assisted Protocol), for devices to exchange protocol related information between each other in both active and on-demand manners. It allow devices to request specific information from other devices and receive replies to the requested data. It also allows actively transmission of information without request to inform other devices to better react w.r.t. network issues.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

Status of This Memo

This Internet-Draft will expire on 12 September 2023.

Copyright Notice

Table of Contents

1. Introduction

A healthy control plane, providing network connectivity, is the foundation of a well-functioning network. There have been rich routing and signaling protocols designed and used for IP networks, such as IGP (ISIS,OSPF), BGP, LDP, RSVP-TE and so on. The health issues of these protocols, such as neighbor/peer disconnect/set up failure, LSP set up failure, route flapping and so on, have been devoted with ongoing efforts for diagnosing and remediation.

1.1. Motivation

The distributive protocol troubleshooting approach is typically realized through manual per-device check. It's both time- and labor-consuming, and requires NOC experience of the operators. Amongst all, localizing the target device is usually the most diffcult and time-consuming part. For example, in the case of route loop, operators first log in a random deivce that reports TTL alarms, and then check the looped route in the Forwarding Information Base (FIB) and/or the Routing Information Base (RIB). It requires device by device check, as well as manul data correlation, to pin point to the exact responsible device, since the information retrival and analysis of such distributive way is fragmented. In addition, the low efficiency and manul troubleshooting activities may further impact new network services and/or enlarge affected areas.

The centralized network OAM, by collecting network-wide data from devices, enables automatic routing protocol troubleshooting. Date collection protocols, such as SNMP (Simple Network Management Protocol) [RFC1157], NETCONF (Network Configuration Protocol) [RFC6241], and (BMP) [RFC7854], can provide various information retrival, such as network states, routing data, configurations and so on. Such centrazlized way relies on the existence of a centralized server/controller, which is not supported by some legacy networks. What's more, even with the existence of a centralized server/controller, it can only collect the data within its own management domain, while the cross-domain data are not available due to independent managment of different ISPs. Thus, the lack of such information may lead to troubleshooting failure. In addition, centralized approaches may suffer from high network bandwidth and CPU computation consumptions.

Another way of protocol troubleshooting is utilzing the protocol itself to convey diagnosing information. For example, some reason codes are carried in the Path-Err/ResvErr messages of RSVP-TE, so that to other nodes may know the why the tunnel fails to be set up. Such approaches is semi-distributive and semi-centralized. It does not rely on the deployment of a centralized server, but still gets partial global view of the network. However, there still requires non-trivial augementation works to existing routing protocols in order to support troubleshooting. This then raises the question that whether such non-routing data is suitable to be carried in these routing protocols. The extra encapsulation, parsing and analyzing work for the non-routing data would further slow down the network convergence. Thus, it's better to separate the routing and non-routing data transmission as well as data parsing. In addition, coexisting with legacy devices may cause interop issues. Thus, relying on augumenting existing routing protocols without network-wide upgrading may not only fail to provide the truobleshooting benefit, but further affect the operation of the existing routing system. What's more, the failure of routing protocol instance would lead to the failure of diagnosing itself. All in all, it's reasonable to separate the protocol diagnosing data generation/encapsulation/transmission/parsing from the protocol itself.

This document proposes a new protocol, called the PASP (Protocol assisted Protocol), for devices to exchange protocol related information between each other. It allows both active and on-demand data exchange. Considering that massiveness of protocol/routing related data, the intuitive of designing PASP is not to exchange the comprehensive protocol/routing status between devices, but to provide very specific information required for fast troubleshooting. The benefits of such a semi-distributive and semi-centralized approach are summarized as follows:

1. It facilitates automatic troubleshooting without requiring manul device by device check.

2. It allows individual device to have a more global view by requesting data from other devices.

3. It does not rely on the deployement of a centralized server/ controller.

4. It passes the data collection boundary set by different management domains by cross-domain data exchange between devices.

5. It relieves the bandwidth pressure of network-wide data collection, and the processing pressure of the centralized server.

6. It does not affect the running of existing routing protocols.

1.2. PASP Use cases

PASP allows both data request/reply and data notification between devices. PASP speakers use the exchanged PASP data to help quickly localize the network issues.

1.2.1. Use Case 1: BGP Route Oscillation

A BGP route oscillation can be caused by various reasons, and usually leaves network-wide impact. In order to find the root cause and take remediation actions, the first step is to localize the oscillation source. In this case, a BGP speaker can send a PASP Request Message to the next hop device of the oscillating route asking " Are you the oscillation source?". If the BGP speaker is the oscillation source, possiblly knows by running a device diagnosing system, replies with a PASP Reply Message saying that "I'm the oscillation source!" to the device who sends the PASP Request Message. If the BGP speaker is not the oscillation source, it further asks the same question with a PASP Request Message to its next hop device of the oscillating route. This request and reply process continues util the request has reached the oscillation source. The source device then sends a PASP Reply Message to tell its upstream device along the PASP request path that " I am the oscillation source!", and then "xx is the oscillation source!"

information is further sent back hop by hop to the device who originates the request.

1.2.2. Use Case 2: RSVP-TE Set Up Failure

The MPLS label switch path set up, either using RSVP-TE or LDP, may fail due to various reasons. Typical troubleshooting procedures are to log in the device, and then check if the failure lies on the configuration, or path computation error, or link failure. Sometimes, it requires the check of multiple devices along the tunnel. Certain reason codes can be carried in the Path-Err/ResvErr messages of RSVP-TE, while other data are currently not supported to be transmitted to the path ingress/egress node, such as the authentication failure. Using PASP, the device, which is reponsible for the tunnel set up failure, can send the PASP Notification Message to the ingress device, and possibly with some reason codes so that the ingress device can not only localize the target device but also the root cause.

1.2.3. Use Cases 3: Peer Disconnection (for IGP/BGP/LDP/BFD)

In a peer disconnected situation, a typical troubleshooting procedure is to login to both devices and check the error log of specific protocols. This is quite difficult if those devices are far away from each other, either geometrically or administratively. Using PASP, a device that suffers the disconnection could send a PASP Request to the disconneted peer. The device that triggers the disconnection could send a PASP Reply with the reason of disconnection, including manual shutdown, TCP down and so on.

1.2.4. Use Cases 4: Detecting Route Interruption

Route Interruption could occur randomly on devices. It is typically short-lived and threfore difficult to be catched in time. Often, when an O&M personnel reaches to the device, the interruption had recovered and the real causes remain uncovered. The distance problem could also exist in this scenario. PASP could collecting route change history, so that rapid route interruptions can be detected and logged. Certain data could be fetched up on request, with a PASP Request message from a trusted source.

1.2.5. Use Cases 5: BGP Route No-advertise

After a BGP peer relationship is established, expected routes may not be advertised or may be withdrawn unexpectedly. Troubleshooting for these situations need the O&M personnel login to both devices and check the status of the routes and peer to determine the cause. Due to the time validity issue, O&M personnel may need to check both BGP speaker simultaneously. Using PASP, device that suffers from a no-advertise situation could send a PASP Request with specific IP address. Receiver could send an PASP Reply with reason of no-advertise, including egress filters, no-advertise attribute and so on.

1.2.6. Use Cases 6: Route Abnormal

Traffic interruption caused by abnormal routes is a common network problem, which could have a great impact on users. It usually takes a lot of time and energy for O&M personnel to locate the device where traffic is interrupted, especially on a large-scale network. With PASP depolyed, an O&M personnel could send a PASP Request message with the specific IP address on any connected device to another device. Receiver could send a PASP Reply with situation codes including nexthop unreachable, outbound interface down, suppression and others.

1.2.7. Use Cases 7: Management protocol failures

Many North-South management protocols, such as SNMP and SSH, are widely used to manage devices. The failure of the management protocol itself could result in a login error or others, which could bring great difficulties in O&M. An O&M personnel could send a PASP Request on a neighbour device to the target device, asking for the reason of failure of a management protocol. In this scenario, PASP can provide another channel for obtaining O&M information of management protocols.

1.2.8. Use Cases 8: Collecting other O&M Events

PASP could record O&M events, such as IP-address conflict, memory leak and so on. Certain data could be fetched up on request, with a PASP Request message from a trusted source. Therefore O&M personnel could obtain those information without repeatedly checking every device in the network.

2. Terminology

IGP: Interior Gateway Protocol

IS-IS: Intermediate System to Intermediate System

OSPF: Open Shortest Path First

BGP: Boarder Gateway Protocol

BGP-LS: Boarder Gateway Protocol-Link State

MPLS: Multi-Protocol Label Switching

RSVP-TE: Resource Reservation Protocol-Traffic Engineering

LDP: Label Distribution Protocol

BMP: BGP Monitoring Protocol

LSP: Link State Packet

IPFIX: Internet Protocol Flow Information Export

PASP: Protocol assisted Protocol

UDP: User Datagram Protocol

3. PASP Overview

3.1. PASP Encapsulation

PASP uses UDP as its transport protocol, which is connectionless. The reason that UDP is selected over TCP is because PASP is intended for on-demand communications. The PASP packet is defined as follows. This document requires the assignment of a User Port registry for the UDP Destination Port.

```
+-------------+------------+------------+------------+------------+
| ETH. Header |  IP Header  | UDP Header  |  PASP Header| PASP Payload|
+-------------+------------+------------+------------+------------+
```

Figure 1. Encapsulation in UDP

3.2. PASP Speaker and PASP Agent

This document uses PASP speakers to refer to routing devices that communicate with each other using PASP. PASP speakers SHOULD be implemented with a supporting module (or multiple modules) to receive, parse, analyze, generate, and send PASP messages. For example, a BGP diagnosing module used for BGP related PASP message handling functions as a PASP agent. A PASP Agent is the union of multiple such modules regarding different protocols, or one module for all protocols. Such supporting module is called PASP Agent in this document. PASP Agent, standalone, SHOULD be able to provide protocol troubleshooting capability with local information. Enabling PASP exchange capability, PASP agent gains information from remote PASP speakers to improve diagnosing accuracy . The primary function of PASP is to provide a unfied tunnel for protocol diagnosing information exchange without augumenting each specific protocol.

3.3. PASP Event

A PASP Event is referred to as the a troubleshooting instance running within a PASP Agent. A PASP Agent may instantiate one or multiple PASP Events for each protocol at the same time depending on the configured troubleshooting triggering condition. For example, an PASP Event is intiated automatically when device CPU is over high, or manually with related command line input from a device operator. Once a PASP Event is generated, corresponding PASP processes are to be called on demand. Notice, the initiation of PASP Capability Negotiation does not require the existance of a PASP Event.

3.4. Summary of Operation

The communications between two PASP speakders should follow three major processes, i.e., the Capability Negotiation Process, the Request and Reply Process, and the Notification Process. This document defines 5 PASP Message types, i.e., Negotiation Message, Request Message, Reply Message, Notification Message, and ACK Message, which are used in the above PASP processes.

3.4.1. PASP Capability Negotiation Process

The purpose of the Capability Negotiation process is to inform two PASP speakers of each other's PASP capabilties. The PASP capability indicates, for which specific protocol(s), that PASP supports its/ their diagnosing information exchange. The process can be further divided into three procudures: 1) PASP Peering Relations Establish process, 2) PASP Capability Enabling Notification Process, 3) PASP Capability Disabling Notification Process. The Capability Negotiation Process is realized by the exchange of PASP Capability Negotiation Message, which is defined in Section 4.

Although PASP is connectionless, a successful PASP Peering Relations Establish Process is required to be successfully performed before any other PASP process. This process can be initiated by either the local or remote PASP speaker through sending out a PASP Capability Negotiation Message. The Negotiation Message may or may not require an ACK Message, as indicated in the Negotiation Message. A successful Peering is established if both PASP speakers have correctly received the other speaker's Capability Negotiation Message. After a successful negotiation, two PASP speakers can exchange any PASP Message on-demand. The PASP Capability Enabling Notification Process is used to inform the PASP peer its newly supported capability, which can be intiated by the PASP speaker at any moment after a PASP Peering is established with the respective PASP Peer. The PASP Capability Disabling Notification Process is used to inform the PASP peer its newly unsupported capability, which can be intiated by the PASP speaker at any moment after a PASP Peering is established with the respective PASP Peer.

3.4.2. PASP Request and Reply Process

The purpose of the PASP Request and Reply Process is to acquire information needed by a PASP speaker from other PASP speakers for a specific PASP Event. The Request and Reply Messages can be customized for different events. The process is triggered by the instantiation of a PASP Event, and starts with sending a Request Message to a target PASP peer. The target PASP peer is selected by the PASP agent regarding the current PASP Event, which is out of the scope of this document. The remote PASP speaker, after receiving the Request Message, sends out a Reply Message to the request sender. ACK is required or not as indicated in the Message Flag.

One Request Message received at the local PASP speaker from a PASP peer may further results in a new Request Message generation regarding a third PASP speaker, if the local PASP speaker does not have the right Reply to this PASP peer. This local PASP speaker does not send Reply Message to the requesting PASP peer until it receives a new Reply Message from this third PASP speaker. So the whole process In order to avoid Request/Reply loops, a Residua Hop value is used to limit the Request/Reply rounds.

3.4.3. PASP Notification Process

The Notification Process is used by a PASP speaker voluntarily to notify other PASP speakers of certain information regarding a PASP Event. The process is triggered by the instantiation of a PASP Event, and starts with sending a Notification Message to one or multiple target PASP peer(s). The target PASP peer(s) is/are selected by the PASP agent regarding the current PASP Event, which is out of the scope of this document. The Notification Message may or may not require an ACK Message, as indicated in the Notification Message.

4. PASP Message Format

4.1. Common Header

The common header is encapsulated in all PASP messages. It is defined as follows.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+---------------+---------------+-------------------------------+
|V|  Flag       |   Msg. Type   |             Length            |
+---------------+---------------+-------------------------------+
+                    Peer Address (16 bytes)                    +
~                                                               ~
+-------------------------------+-------------------------------+
|        Msg. Sequence          |
+-------------------------------+
```

Figure 2. PASP Common Header

*Flag (1 byte): The V flag indicates that the source IP address is an IPv6 address. For IPv4 address, this is set to 0.

*Message Type (1 byte): This indicates the PASP message type.The following types are defined, and listed as follows.

-Type = TBD1: Capability Negotiation Message. It is used for two devices to inform each other of the capabilties they support and no longer support.

-Type = TBD2: Request Message.

-Type = TBD3: Reply Message.

-Type = TBD4: Notification Message.

-Type = TBD5: ACK Message. It is used to confirm to the local device that the remote device has received a previous sent PASP message, which can be either a Negotiation Message, a Request Message, a Reply Message or an Notification Message.

*Length (2 bytes): Length of the message in bytes, including the Common Header and the following Message.

*Souece IP Address (16 bytes): It indicates the IP address who initiates the PASP message. It is 4 bytes long if an IPv4 address is carried in this field (with the 12 most significant bytes zero-filled) and 16 bytes long if an IPv6 address is carried in this field.

*Message Sequence (2 bytes): It indicates the sequence number of each PASP message.

4.1.1. Capability Negotiation Message

The Negotiation Message is used in the PASP Capability Negotiation Process. It is defined as follows.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+---------------------------------+-------------------------------+
|     Version   |A|E|   Flag     |
+---------------------------------+-------------------------------+
|                          Protocol Capacity                      |
+-----------------------------------------------------------------+
```

Figure 3. PASP Negotiation Message

*Version (1 byte): It indicates the PASP version. The current version is 0.

*Flags (1 bytes): Two flag bits are currently defined.

-The A bit is used to indicate if an ACK Message from the remote PASP speaker is required for each Negotiation Message sent. If an ACK is required, then the A bit SHOULD be set to "1", and "0" otherwise.

-The E bit is used to indicate the enabling/disabling of the capabilities that carried in the Protocol Capability field. If the local device wants to inform the remote device of enabling one or more capabilities, the E bit SHOULD be set to "1". If the local device wants to inform the remote device of disabling one or more capabilities, the E bit SHOULD be set to "0".

*Protocol Capability (4 bytes): It is 4-byte bitmap that indicates the capability of inforamtion exchange regarding various protocols. Each bit represents one protocol. The following protocol capability is defined (from the rightmost bit).

-Bit 0: ISIS

-Bit 1: OSPF

-Bit 2: BGP

-Bit 3: LDP

4.2. Request Message

The Request Message is used for the local device to request specific data regarding one specific protocol or application from the remote device. It MUST be sent after a successful Capability Negotiation Process (described in Section 5.1), and the requested protocol/ application MUST be supported by both the local and remote devices, as indicated in the Negotiation Messages exchanged between the local and remote devices. It is defined as follows.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+---------------+---------------+-------------------------------+
|A|    Flag     |   Prot. Capb. |          Event ID             |
+-------------------------------+-------------------------------+
|  Res. Hop     |
+---------------+-----------------------------------------------+
+                          Request Data                         +
~                                                               ~
+---------------------------------------------------------------+
```

Figure 4. PASP Request Message

*Flags (1 byte): It is currently reserved. The A bit is used to indicate if an ACK Message from the remote PASP speaker is required for each Request Message sent. If an ACK is required, then the A bit SHOULD be set to "1", and "0" otherwise.

*Capability (1 byte): It represents the bit index of the protocol, which the Request Message is requesting data for.

*Event ID (2 bytes): It indicates the event number that this Request message is regarding.

*Residua hop (1 byte): it indicates the residua Request hops of the current PASP Event. It is reduced by 1 at each PASP speaker when generating a further PASP Request to a third PASP speaker.

*Request Data (Variable): Specifies information of the data that the local device is requesting. The specific format remains to be determined per each protocol, as well as each use case.

4.3. Reply Message

The Reply Message is used to carry the information that the local device requests from the remote device through the Request Message. It is defined as follows.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+---------------+---------------+-------------------------------+
|A|    Flag     |   Prot. Capb. |          Event ID             |
+---------------+---------------+-------------------------------+
+                          Reply Data                           +
~                                                               ~
+---------------------------------------------------------------+
```

Figure 5. PASP Reply Message

*Flags (1 byte): It is currently reserved. The A bit is used to indicate if an ACK Message from the remote PASP speaker is required for each Reply Message sent. If an ACK is required, then the A bit SHOULD be set to "1", and "0" otherwise.

*Capability (1 byte): It represents the bit index of the protocol, which the Reply Message is replying data for.

*Event ID (2 bytes): It indicates the event number that this Reply message is regarding.

*Reply Data (Variable): Specifies information of the data that the local device is replying. The specific format remains to be determined per each protocol, as well as each use case.

4.4. Notification Message

The Notification Message is used to carry the information that the local device sends to the remote device.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+---------------+---------------+-------------------------------+
|A|    Flag     |   Prot. Capb. |          Event ID             |
+---------------+---------------+-------------------------------+
+                     Notification Data                         +
~                                                               ~
+---------------------------------------------------------------+
```

Figure 6. PASP Notification Message

*Flags (1 byte): It is currently reserved. The A bit is used to indicate if an ACK Message from the remote PASP speaker is required for each Notification Message sent. If an ACK is required, then the A bit SHOULD be set to "1", and "0" otherwise.

*Capability (1 byte): It represents the bit index of the protocol, which the Notification Message is notifying for.

*Event ID (2 bytes): It indicates the event number that this Notification Message is regarding.

*Notification Data (Variable): Specifies information of the data that the local device is notifying. The specific format remains to be determined per each protocol, as well as each use case.

4.5. *ACK Message

The ACK Message is used to confirm that the remote device has received a PASP Message with the A bit set to "1". The ACK Message includes only the PASP Common Header. The Msg. Sequence MUST be set to the sequence number carried in the received PASP message, which requires this ACK.

5. PASP Operations

The PASP operations include the following 3 major processes, the Capability Negotiation Process, the Data Request and Reply Process, and the Data Notification Process.

5.1. Capability Negotiation Process

5.1.1. PASP Peering Relation Establish Process

A successful PASP Peering relation MUST be Established between two PASP speakers before any other PASP process.

As the first step, a Capability Negotiation Message can be initiated at any time by a PASP speaker,as long as the target PASP peer is IP reachable. It usually companies the establishment of neighboring/ peering relation between two routing devices. The "A" bit in the Negotiation Message MUST be set as 1 during the PASP Peering Establish Process, meaning ACK required. The "E" in the Negotiation Message MUST be set to 1 during this process, meaning the capabilities indicated in the Protocol Capability field are enabled by default. The Protocol Capability field SHOULD indicate all the protocol capabilities that are supported by the local PASP Agent and currently enabled. After the first Negotiation Message is sent, the local device SHUOLD wait for the ACK Message from the remote device for a certain time period before taking further actions, and if no ACK Message is received within this time frame, the local device SHOULD resend the Negotiation Message to the remote device. The waiting period can be configured locally. This send and wait process CAN be repeated for at most 3 times before receiving a ACK Message from the remote device. If after 3 times of resending the Negotiation Message, still no ACK received, then this peering establishment is treated as unsuccessful.

The next step for the local PASP speaker is to wait for the Negotiation Message from the remote PASP speaker. If no Negotiation Message is received from the remote PASP speaker within a time frame after its own Negotiation Message is sent , the local PASP speaker CAN resend the Negotiation Message. This time frame is also configured locally. This send and wait process CAN be repeated for at most 3 times before receiving a Negotiation Message from the remote PASP speaker. If after 3 times of resending the Negotiation Message, still no Negotiation Message received, then this negotiation is treated as unsuccessful. If a Negotiation Message is received and parsed correctly, an ACK MUST be sent to the remote PASP speaker.

Once an ACK Message and a Negotiation Message are received from the remote PASP speaker and correctly parsed, a PASP Peering relation is considered as successfully established. The local PASP speaker maintains locally the protocol capabilities of the remote PASP speaker, and uses them during other PASP processes.

5.1.2. PASP Capability Enabling Notification Process

Once the PASP Peering relation is set up between two PASP speakers, they become PASP peers. Thereafter, any PASP speaker supports a new protocol capability, it SHOULD call the Capability Enabling Notification Process to inform all its PASP peers.

When the local PASP speaker initates a PASP Capability Enabling Notification Process: The "A" bit in the Negotiation Message MUST be set as 1 during the PASP Capability Enabling Notification Process, meaning ACK required. The "E" in the Negotiation Message MUST be set to 1 during this process, meaning the capabilities indicated in the Protocol Capability field are enabled. The Protocol Capability field SHOULD indicate all the protocol capabilities that are supported by the local PASP Agent and currently enabled. After the Negotiation Message is sent, the local PASP speaker SHUOLD wait for the ACK Message from the PASP peer for a certain time period before taking further actions, and if no ACK Message is received within this time frame, the local device SHOULD resend the Negotiation Message to the remote device. The waiting period can be configured locally. This send and wait process CAN be repeated for at most 3 times before receiving a ACK Message from the remote device. If after 3 times of resending the Negotiation Message, still no ACK received, then this Capability Enabling Notification Process is treated as unsuccessful. This process MAY be intiated at another time thereafter. If a ACK is received, the Capability Enabling Notification Process is considered successful.

When a PASP peer initates a PASP Capability Enabling Notification Process: The local PASP speaker, after receiving the PASP

Negotiation Message and correctly parsing it, sends out an ACK. This Capability Enabling Notification Process is considered successful. The local PASP speaker updates the capability status maintained accordingly.

5.1.3. PASP Capability Disabling Notification Process

Whenever a PASP speaker disables a PASP capability, it SHOULD initiate a PASP Capability Disabling Notification Process to inform all its PASP peers.

When the local PASP speaker initates a PASP Capability Disabling Notification Process: The "A" bit in the Negotiation Message MUST be set as 1 during the PASP Capability Disabling Notification Process, meaning ACK required. The "E" in the Negotiation Message MUST be set to 0 during this process, meaning the capabilities indicated in the Protocol Capability field are disabled. The Protocol Capability field SHOULD indicate all the protocol capability that is disabled. After the Negotiation Message is sent, the local PASP speaker SHUOLD wait for the ACK Message from the PASP peer for a certain time period before taking further actions, and if no ACK Message is received within this time frame, the local device SHOULD resend the Negotiation Message to the remote device. The waiting period can be configured locally. This send and wait process CAN be repeated for at most 3 times before receiving a ACK Message from the remote device. If after 3 times of resending the Negotiation Message, still no ACK received, then this Capability Disabling Notification Process is treated as unsuccessful. This process MAY be intiated at another time thereafter.

When a PASP peer initates a PASP Capability Disabling Notification Process: The local PASP speaker, after receiving the PASP Negotiation Message and correctly parsing it, sends out an ACK. This Capability Disabling Notification Process is considered successful. The local PASP speaker updates the capability status maintained accordingly.

5.2. PASP Request and Reply Process

When a local PASP Event triggers a PASP Request and Reply Process, the local PASP speaker initates a Request Message, and send to a target PASP peer as indicated by PASP Agent per this PASP Event. This local PASP speaker is called the Request and Reply Process Starter. It sets the Residua Hop as the maximum number of Request/Reply rounds (e.g., 10) it will wait in order to receive the final Reply. The Event ID and the Request are set by the local PASP Agent. The A bit of the Request Message MUST be set to "1" (i.e., ACK is required). The local device waits for the ACK Message from the remote device for a certain time period before taking further actions, and if no ACK Message is received within this time frame, the local device SHOULD resend the Request Message to the remote device. The waiting period can be configured locally. This send and wait process CAN be repeated for at most 3 times before receiving a ACK Message from the remote device. If after 3 times of resending the Request Message, still no ACK received, then this Request and Reply Process is treated as unsuccessful. If ACK received, the local device waits for the Reply Message. If no Reply Message is received from the remote device within a time frame, the local device can resend the Request Message. This send and wait process CAN be repeated for at most 3 times before receiving a Reply Message from the remote device. If after 3 times of resending the Request Message, still no Reply Message received, then this Request and Reply Process is treated as unsuccessful. The waiting period can be configured locally, and SHOULD take into consideration of the Residua Hop value. If the Request and Reply Process Starter receives the Reply Message within the time frame, and the Event ID is matched to the local PASP Event, the PASP Request and Reply Process is considered as successful.

When a local PASP speaker receives a Request Message from its PASP peer (i.e., it is not the Pequest and Reply Process Starter), it sends back an ACK Message. With the received Request Message, a new PASP event it instantiated at the local PASP Agent. The PASP event triggers the troubleshooting analysis of the received Request Message, and then generate the Reply Message if the Reply condition is met, or generate a new Request Message when the Reply condition is not met. The Reply condition and the troubleshooting analysis of the PASP Agent is out of the scope of this document.

If the Reply condition is met, the local PASP speaker is called the Request and Reply Process Terminator. It generates the Reply Message and send the message back to the requesting PASP peer. The Event ID is set to be the same as the Event ID of the received Request Message. The Reply Data is set by the local PASP Agent per this generated event. The A bit of the Reply Message MUST be set to "1" (i.e., ACK is required). The local device waits for the ACK Message from the remote device for a certain time period before taking further actions, and if no ACK Message is received within this time frame, the local device SHOULD resend the Reply Message to the remote device. The waiting period can be configured locally. This send and wait process CAN be repeated for at most 3 times before receiving a ACK Message from the remote device. If after 3 times of resending the Request Message, still no ACK received, then this Request and Reply Process is treated as unsuccessful.

If the Reply condition is not met, the local PASP speaker is called the Request and Reply Process mid-handler. It generates a new Request Message and send the message to a third PASP speaker per indicated by the local PASP Agent per this generated event. In the new generated Request Message, the Residua Hop value by MUST be reduced by 1. The A bit of the Request Message MUST be set to "1" (i.e., ACK is required). The local device waits for the ACK Message from the remote device for a certain time period before taking further actions, and if no ACK Message is received within this time frame, the local device SHOULD resend the Request Message to the remote device. The waiting period can be configured locally. This send and wait process CAN be repeated for at most 3 times before receiving a ACK Message from the remote device. If after 3 times of resending the Request Message, still no ACK received, then this Request and Reply Process is treated as unsuccessful. If ACK received, the local device waits for the Reply Message. If no Reply Message is received from the remote device within a time frame, the local device can resend the Request Message. This send and wait process CAN be repeated for at most 3 times before receiving a Reply Message from the remote device. If after 3 times of resending the Request Message, still no Reply Message received, then this Request and Reply Process is treated as unsuccessful. The waiting period can be configured locally, and SHOULD take into consideration of the Residua Hop value. If the local device receives the Reply Message within the time frame, it generates a new Reply Message and sends back to it requesting PASP peer. The Event ID of the new Reply Message is set to be the same as the Event ID of the received Request Message.

5.3. PASP Notification Process

When a local PASP Event triggers a PASP Notification Process, the local PASP speaker initates a Notification Message. The target PASP peer(s) is/are selected by the PASP agent regarding the current PASP Event, which is out of the scope of this document. The Notification Message may or may not require an ACK Message, as indicated in the Notification Message. If the A bit is set to 1 (meaning ACK required), the local device waits for the ACK Message from the remote device for a certain time period before taking further actions, and if no ACK Message is received within this time frame, the local device SHOULD resend the Notification Message to the remote device. The waiting period can be configured locally. This send and wait process CAN be repeated for at most 3 times before receiving a ACK Message from the remote device. If after 3 times of resending the Request Message, still no ACK received, then this Request and Reply Process is treated as unsuccessful. The waiting period can be configured locally. If ACK is received within the time frame, the Notification Process is considered to be successful. If the A bit is set to 0 (meaning no ACK required), after sending the Notification Message, the Notification Process is considered successful.

6. PASP Error Handling

When any PASP process is unsuccessful, information is recorded or not by local PASP Agent. No further action is taken.

7. Discussion

In addition to the preceding message definition and process description, the security and reliability requirements of the PASP need to be considered. There are two possible options to implement PASP.

- Option 1: PASP is developed independently as a new protocol.

- Option 2: PASP reuses the existing protocol Generic Autonomic Signaling Protocol(GRASP) [RFC8990] .

Option1:

1. Definition of the Message Format and Interaction Process: It can be defined independently in the PASP.

2. Reliability: The transmission mode of PASP is based on UDP mainly considering that the collected information is the auxiliary information to help locate the protocol fault, and the information loss has no impact on the service. In addition, if TCP mode is adopted, the resource consumption of the device may be large, especially when there area large number of neighbors. If it is considered that PASP must ensure reliability, it can done in the application layer, such as adding the sequence number to the message.

3. Security: MD5 authentication can be introduced for PASP security.

Option2:

ANIMA GRASP is a signaling protocol used for dynamic peer discovery, status synchronization, and parameter negotiation between AS nodes or AS service agents. GRASP specifies that unicast packets must be transmitted based on TCP, and multicast packets (Discovery and Flood) must be transmitted based on UDP.

1. Message format and interaction process: PASP can reuse the defined messages and procedures of the GRASP. Messages defined in the PASP include Capability Negotiation Message, Request Message, Reply Message, and Negotiation Message. These message types are also defined in GRASP.

2. Reliability: TCP mode of GRASP can be used to ensure reliability for PASP. But there may be some challenges for the equipment resources.

3. Security: Autonomic Control Plane(ACP) [RFC8994] can be reused.

8. Security Considerations

TBD

9. IANA Considerations

TBD

10. Contributors

We thank Jiaqing Zhang (Huawei), Tao Du (Huawei) and Lei Li (Huawei) for their contributions.

11. Acknowledgments

12. References

12.1. Normative References

[RFC1191] Mogul, J. and S. Deering, "Path MTU discovery", RFC 1191, DOI 10.17487/RFC1191, November 1990, <https://www.rfc-editor.org/info/rfc1191>.

[RFC1195] Callon, R., "Use of OSI IS-IS for routing in TCP/IP and dual environments", RFC 1195, DOI 10.17487/RFC1195, December 1990, <https://www.rfc-editor.org/info/rfc1195>.

[RFC1213] McCloghrie, K. and M. Rose, "Management Information Base for Network Management of TCP/IP-based internets: MIB-II", STD 17, RFC 1213, DOI 10.17487/RFC1213, March 1991, <https://www.rfc-editor.org/info/rfc1213>.

[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>.

[RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, DOI 10.17487/RFC3209, December 2001, <https://www.rfc-editor.org/info/rfc3209>.

[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>.

[RFC7752] Gredler, H., Ed., Medved, J., Previdi, S., Farrel, A., and S. Ray, "North-Bound Distribution of Link-State and Traffic Engineering (TE) Information Using BGP", RFC 7752, DOI 10.17487/RFC7752, March 2016, <https://www.rfc-editor.org/info/rfc7752>.

[RFC7854] Scudder, J., Ed., Fernando, R., and S. Stuart, "BGP Monitoring Protocol (BMP)", RFC 7854, DOI 10.17487/RFC7854, June 2016, <https://www.rfc-editor.org/info/rfc7854>.

[RFC8641] Clemm, A. and E. Voit, "Subscription to YANG Notifications for Datastore Updates", RFC 8641, DOI 10.17487/RFC8641, September 2019, <https://www.rfc-editor.org/info/rfc8641>.

[RFC8990] Bormann, C., Carpenter, B., Ed., and B. Liu, Ed., "GeneRic Autonomic Signaling Protocol (GRASP)", RFC 8990, DOI 10.17487/RFC8990, May 2021, <https://www.rfc-editor.org/info/rfc8990>.

[RFC8994] Eckert, T., Ed., Behringer, M., Ed., and S. Bjarnason, "An Autonomic Control Plane (ACP)", RFC 8994, DOI 10.17487/RFC8994, May 2021, <https://www.rfc-editor.org/info/rfc8994>.

12.2. References

[I-D.brockners-inband-oam-requirements]
Brockners, F., Bhandari, S., Dara, S., Pignataro, C., Gredler, H., Leddy, J., Youell, S., Mozes, D., Mizrahi, T., <>, P. L., and remy@barefootnetworks.com, "Requirements for In-situ OAM", Work in Progress, Internet-Draft, draft-brockners-inband-oam-requirements-03, 13 March 2017, <https://datatracker.ietf.org/doc/html/draft-brockners-inband-oam-requirements-03>.

[I-D.song-ntf] Song, H., Zhou, T., Li, Z., Fioccola, G., Li, Z., Martinez-Julia, P., Ciavaglia, L., and A. Wang, "Toward a Network Telemetry Framework", Work in Progress, Internet-Draft, draft-song-ntf-02, 2 July 2018, <https://datatracker.ietf.org/doc/html/draft-song-ntf-02>.

[RFC1157] Case, J., Fedor, M., Schoffstall, M., and J. Davin, "Simple Network Management Protocol (SNMP)", RFC 1157, DOI 10.17487/RFC1157, May 1990, <https://www.rfc-editor.org/info/rfc1157>.

[RFC3988] Black, B. and K. Kompella, "Maximum Transmission Unit Signalling Extensions for the Label Distribution Protocol", RFC 3988, DOI 10.17487/RFC3988, January 2005, <https://www.rfc-editor.org/info/rfc3988>.

Authors' Addresses

Zhenbin Li
Huawei
156 Beiqing Rd
Beijing
China

Email: lizhenbin@huawei.com

Shuanglong Chen
Huawei
156 Beiqing Road
Beijing,100095
China

Email: chenshuanglong@huawei.com

Zhen Tan
Huawei
156 Beiqing Rd
Beijing
China

Email: tanzhen6@huawei.com

Yingzhen Qu
Futurewei

Email: yingzhen.qu@futurewei.com

Yunan Gu
Huawei
156 Beiqing Rd
Beijing
China

Email: guyunan@huawei.com