Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 6, 2021

Z. Li
S. Chen
Y. Gu
Huawei

November 02, 2020

Protocol Assisted Protocol (PAP)
draft-li-rtgwg-protocol-assisted-protocol-03

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) RFC7854 [RFC7854], 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 PAP (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

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 PAP (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 PAP 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 dtata 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. PAP Usage Use cases

PAP allows both data request/reply and data notification between devices. PAP speakers use the exchanged PAP data to help fast 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 PAP 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 PAP Reply Message saying that "I'm the oscillation source!" to the device who sends the PAP Request Message. If the BGP speaker is not the oscillation source, it further asks the same question with a PAP 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 PAP Reply Message to tell its upstream device along the PAP 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 PAP, the device, which is reponsible for the tunnel set up failure, can send the PAP 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.

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

PAP: Protocol assisted Protocol

UDP: User Datagram Protocol

3. PAP Overview

3.1. PAP Encapsulation

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

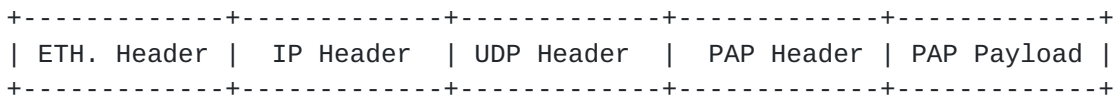

Figure 1. Encapsulation in UDP

3.2. PAP Speaker and PAP Agent

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

3.3. PAP Event

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

3.4. Summary of Operation

The communications between two PAP 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 PAP Message types, i.e., Negotiation Message, Request Message, Reply Message, Notification Message, and ACK Message, which are used in the above PAP processes.

3.4.1. PAP Capability Negotiation Process

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

Although PAP is connectionless, a successful PAP Peering Relations Establish Process is required to be successfully performed before any other PAP process. This process can be initiated by either the local or remote PAP speaker through sending out a PAP 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 PAP speakers have correctly received the other speaker's Capability Negotiation Message. After a successful negotiation, two PAP speakers can exchange any PAP Message on-demand. The PAP Capability Enabling Notification Process is used to inform the PAP peer its newly supported capability, which can be intiated by the PAP speaker at any moment after a PAP Peering is established with the respective PAP Peer. The PAP Capability Disabling Notification Process is used to inform the PAP peer its newly unsupported capability, which can be intiated by the PAP speaker at any moment after a PAP Peering is established with the respective PAP Peer.

3.4.2. PAP Request and Reply Process

The purpose of the PAP Request and Reply Process is to acquire information needed by a PAP speaker from other PAP speakers for a specific PAP Event. The Request and Reply Messages can be customized for different events. The process is triggered by the instantiation of a PAP Event, and starts with sending a Request Message to a target PAP peer. The target PAP peer is selected by the PAP agent regarding the current PAP Event, which is out of the scope of this document.

The remote PAP 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 PAP speaker from a PAP peer may further results in a new Request Message generation regarding a third PAP speaker, if the local PAP speaker does not have the right Reply to this PAP peer. This local PAP speaker does not send Reply Message to the requesting PAP peer until it receives a new Reply Message from this third PAP 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. PAP Notification Process

The Notification Process is used by a PAP speaker voluntarily to notify other PAP speakers of certain information regarding a PAP Event. The process is triggered by the instantiation of a PAP Event, and starts with sending a Notification Message to one or multiple target PAP peer(s). The target PAP peer(s) is/are selected by the PAP agent regarding the current PAP 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. PAP Message Format

4.1. Common Header

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

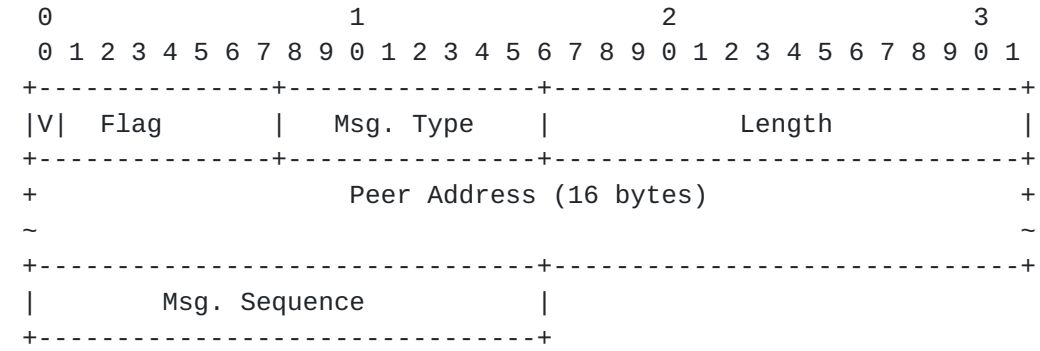

Figure 2. PAP Common Header

o 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 PAP 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 PAP 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 PAP 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 PAP message.

4.1.1. Capability Negotiation Message

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

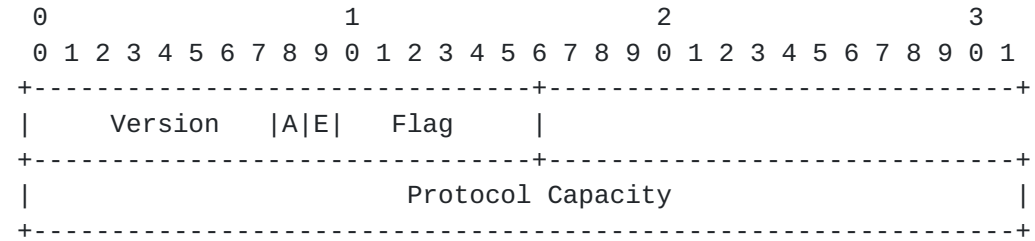

Figure 3. PAP Negotiation Message

- Version (1 byte): It indicates the PAP 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 PAP 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.

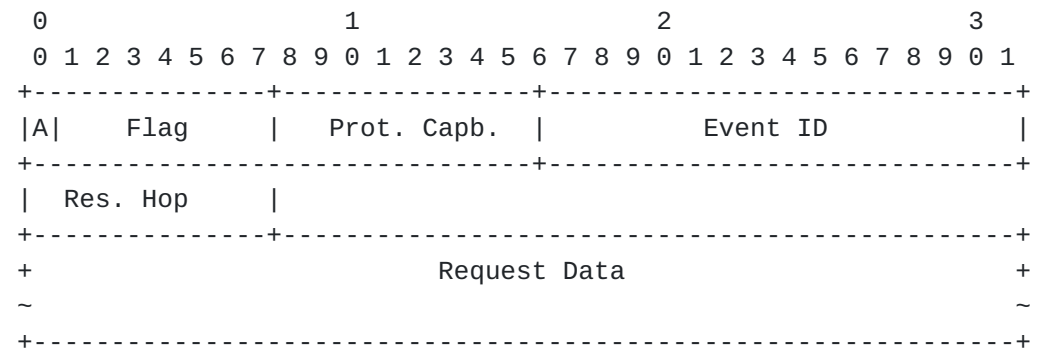

Figure 4. PAP Request Message

- Flags (1 byte): It is currently reserved. The A bit is used to indicate if an ACK Message from the remote PAP 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 PAP Event. It is reduced by 1 at each PAP speaker when generating a further PAP Request to a third PAP 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.

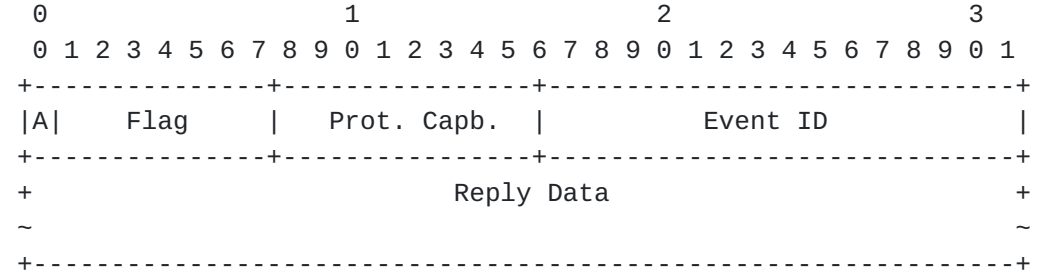

Figure 5. PAP Reply Message

- Flags (1 byte): It is currently reserved. The A bit is used to indicate if an ACK Message from the remote PAP 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.

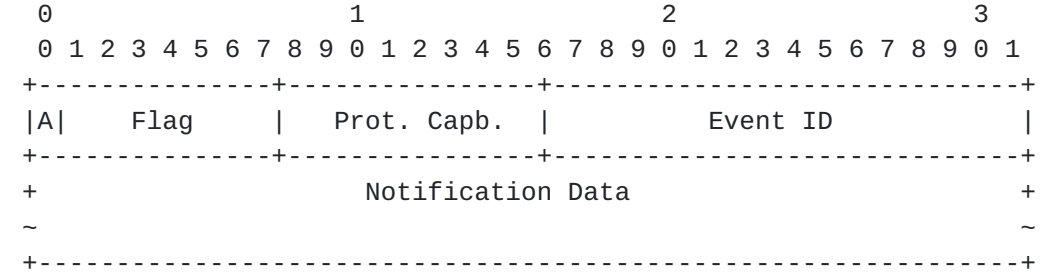

Figure 6. PAP Notification Message

- Flags (1 byte): It is currently reserved. The A bit is used to indicate if an ACK Message from the remote PAP 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 PAP Message with the A bit set to "1". The ACK Message includes only the PAP Common Header. The Msg. Sequence MUST be set to the sequence number carried in the received PAP message, which requires this ACK.

5. PAP Operations

The PAP 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. PAP Peering Relation Establish Process

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

As the first step, a Capability Negotiation Message can be initiated at any time by a PAP speaker,as long as the target PAP 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 PAP 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 PAP 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 PAP speaker is to wait for the Negotiation Message from the remote PAP speaker. If no Negotiation Message is received from the remote PAP speaker within a time frame after its own Negotiation Message is sent , the local PAP 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 PAP 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 PAP speaker.

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

5.1.2. PAP Capability Enabling Notification Process

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

When the local PAP speaker initates a PAP Capability Enabling Notification Process: The "A" bit in the Negotiation Message MUST be set as 1 during the PAP 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 PAP Agent and currently enabled. After the Negotiation Message is sent, the local PAP speaker SHUOLD wait for the ACK Message from the PAP 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 PAP peer initates a PAP Capability Enabling Notification Process: The local PAP speaker, after receiving the PAP Negotiation Message and correctly parsing it, sends out an ACK. This Capability Enabling Notification Process is considered successful. The local PAP speaker updates the capability status maintained accordingly.

5.1.3. PAP Capability Disabling Notification Process

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

When the local PAP speaker initates a PAP Capability Disabling Notification Process: The "A" bit in the Negotiation Message MUST be set as 1 during the PAP 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 PAP speaker SHUOLD wait for the ACK Message from the PAP 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 PAP peer initates a PAP Capability Disabling Notification Process: The local PAP speaker, after receiving the PAP Negotiation Message and correctly parsing it, sends out an ACK. This Capability Disabling Notification Process is considered successful. The local PAP speaker updates the capability status maintained accordingly.

5.2. PAP Request and Reply Process

When a local PAP Event triggers a PAP Request and Reply Process, the local PAP speaker initates a Request Message, and send to a target PAP peer as indicated by PAP Agent per this PAP Event. This local PAP 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 PAP 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 PAP Event, the PAP Request and Reply Process is considered as successful.

When a local PAP speaker receives a Request Message from its PAP 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 PAP event it instantiated at the local PAP Agent. The PAP 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 PAP Agent is out of the scope of this document.

If the Reply condition is met, the local PAP speaker is called the Request and Reply Process Terminator. It generates the Reply Message and send the message back to the requesting PAP 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 PAP 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 PAP speaker is called the Request and Reply Process mid-handler. It generates a new Request Message and send the message to a third PAP speaker per indicated by the local PAP 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 PAP 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. PAP Notification Process

When a local PAP Event triggers a PAP Notification Process, the local PAP speaker initates a Notification Message. The target PAP peer(s) is/are selected by the PAP agent regarding the current PAP 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. PAP Error Handling

When any PAP process is unsuccessful, information is recorded or not by local PAP 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 PAP need to be considered. There are two possible options to implement PAP.

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

- Option 2: PAP reuses the existing protocol Generic Autonomic Signaling Protocol(GRASP) [I-D.ietf-anima-grasp] .

Option1:

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

2. Reliability: The transmission mode of PAP 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 PAP 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 PAP 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: PAP can reuse the defined messages and procedures of the GRASP. Messages defined in the PAP 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 PAP. But there may be challenge for the equipment resources.

3. Security: Autonomic Control Plane(ACP) [I-D.ietf-anima-autonomic-control-plane] can be reused.

8. Security Considerations

TBD

9. IANA

TBD

10. Contributors

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

11. Acknowledgments

12. 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., Lapukhov, P., and r. remy@barefootnetworks.com, "Requirements for In-situ OAM", draft-brockners-inband-oam-requirements-03 (work in progress), March 2017.

[I-D.ietf-anima-autonomic-control-plane]
Eckert, T., Behringer, M., and S. Bjarnason, "An Autonomic Control Plane (ACP)", draft-ietf-anima-autonomic-control-plane-30 (work in progress), October 2020.

[I-D.ietf-anima-grasp]
Bormann, C., Carpenter, B., and B. Liu, "A Generic Autonomic Signaling Protocol (GRASP)", draft-ietf-anima-grasp-15 (work in progress), July 2017.

[I-D.ietf-netconf-yang-push]
Clemm, A. and E. Voit, "Subscription to YANG Datastores", draft-ietf-netconf-yang-push-25 (work in progress), May 2019.

[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", draft-song-ntf-02 (work in progress), July 2018.

[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>.

[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>.

[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>.

[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>.

Authors' Addresses

Zhenbin Li
Huawei
156 Beiqing Rd
Beijing
China

Email: lizhenbin@huawei.com

Shuanglong Chen
Huawei
156 Beiqing Road
Beijing,100095
P.R. China

Email: chenshuanglong@huawei.com

Yunan Gu
Huawei
156 Beiqing Rd
Beijing
China

Email: guyunan@huawei.com