ANIMA WG B. Liu
INTERNET-DRAFT S. Jiang
Intended Status: Standard Track Huawei Technologies
Expires: January 3, 2019 X. Xiao
A. Hecker
Z. Despotovic
MRC, Huawei Technologies
July 2, 2018

Information Distribution in Autonomic Networking
draft-liu-anima-grasp-distribution-06

Abstract

This document discusses the requirement of capability of information distribution among autonomic nodes in autonomic networks. In general, information distribution can be categorized into two different modes: 1) one autonomic node instantly sends information to other nodes in the domain; 2) one autonomic node can publish some information and then some other interested nodes can subscribe the published information. In the former case, information data will be generated and consumed instantly. In the latter case, however, information data shall be stored in the network and retrieved when necessary.

These capabilities are fundamental and basic to a network system and an autonomic network infrastructure (ANI) should consider to integrate them, rather than assisted by other transport or routing protocols (HTTP, BGP/IGP as bearing protocols etc.). Thus, this document clarifies possible use cases and requirements to ANI so that information distribution can be natively supported. Possible options realizing the information distribution function are also briefly discussed.

Copyright and License Notice

Table of Contents

1 Introduction

In autonomic networking, autonomic functions (AFs) running on autonomic nodes utilize autonomic control plane (ACP) to realize various control purposes [RFC7575]. Due to the distributed nature of a network system, AFs need to exchange information constantly, either for control plane signaling, for data plane service or for both.

This document discusses the information distribution capability of an autonomic network. We classify information distribution scenarios into the following two models:

1) An instant communication model where a sender directly connects and sends information data (e.g. control messages, synchronization data and so on) to the receiver(s).

2) An asynchronous communication model where an autonomic node publishes information and any other nodes that are interested in the information can later subscribe that and will be notified if the information become available.

The two communication models should be integrated within the Autonomic Network Infrastructure (ANI) [I-D.behringer-anima-reference-model], rather than assisted by other transport or routing protocols (HTTP, BGP/IGP as bearing protocols etc.). In fact, GRASP already provides some capabilities to support parts of the information distribution function, utilized for stable connectivity as in [I-D.ietf-anima-stable-connectivity-10].

In this document, we analyze possible scenarios of information distribution in autonomic networks (Section 3), and then discuss the technical requirements (Section 4) that an autonomic node has to fulfill. After that, the node behaviors with extensions on current GRASP to realize the information distribution are introduced.

2. Terminology

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].

3. Requirements of Advanced Information Distribution

Information distribution can occur either between two or among multiple network nodes.

- Point-to-point (P2P) Communications:

This is a common scenario in most of network systems. Information are exchanged between two communicating parties from one node to another node. Specifically, the information can be either pushed to the receiver or pulled from a sender. Therefore, we have two sub-cases:

1) One node acquires some information from another one. This is a very common scenario that can already be covered by GRASP.

2) One node actively pushes some information to another one. For example, when some common information are propagated to the network, it is possible that some nodes are sleeping/off-line, so when these nodes get online again, their neighbors could push the information to them immediately.

- One-to-Many Communications:

Some information exchange involve an information source and multiple receivers. This scenario can be divided into two situations:

1) When some information are relevant to all or most of the nodes in the domain, the node that firstly handle the information should use a mechanism to propagate it to all the other nodes. One typical case is the Intent distribution, which is briefly discussed in Section 4.7 of [I-D.ietf-anima-reference-model]. A flood mechanism, which can guarantee the information could reach to every node, is the most proper approach to do this.

2) A more general case is that some information is only relevant to a specific group of nodes belonging to the same sub-domain or sharing the same interests. Then, the information needs to be propagated to the nodes that fit for certain conditions. This could reduce some unnecessary signaling amplification.

Clearly, both of the two scenarios can be directly carried by the instant communication model. Especially, if the information exchange

is simple and short, this can be done instantly. In practice, however, information distribution is not always simple. As examples, in the following cases, a mixture of instant and asynchronous communication models is more appropriate.

1) Long Communication Intervals. The time interval of the communication is not necessarily always short and instant. Advanced AFs may rather involve heavy jobs/tasks when gearing the network, so the direct mode may introduce unnecessary pending time and become less efficient. For example, an AF accesses another AF for a database lookup. Similar use cases include AF migration, AF authentication and authorization. If simply using an instant mode, the AF has to wait until the tasks finish and return. A better way is that an AF instantly sends the request but switches to an synchronous mode, once the jobs are finished, AFs will get notified.

2) Common Interest Distribution. As mentioned, some information are common interests among AFs. For example, the network intent is distributed to network nodes enrolled, which is a typical one-to-many scenario. We can also finish the intent distribution by an instant flooding (e.g. via GRASP) to every network nodes across the network domain. Because of network dynamic, however, not every node can be just ready at the moment when the network intent is flooded. Actually, nodes may join in the network sequentially. In this situation, an asynchronous communication model could be a better choice where every (newly joining) node can subscribe the intent information and will get notified if it is ready (or updated).

3) Distributed Coordination. With computing and storage resources on autonomic nodes, alive AFs not only consumes but also generates data information. For example, AFs coordinating with each other as distributed schedulers, responding to service requests and distributing tasks. It is critical for those AFs to make correct decisions based on local information, which might be asymmetric as well. AFs may also need synthetic/aggregated data information (e.g. statistic info, like average values of several AFs, etc.) to make decisions. In these situations, AFs will need an efficient way to form a global view of the network (e.g. about resource consumption, bandwidth and statistics). Obviously, purely relying on instant communication model is inefficient, while a scalable, common, yet distributed data layer, on which AFs can store and share information in an asynchronous way, should be a better choice.

For ANI, in order to support various communication scenarios, an information distribution module is required, and both instant and

asynchronous communication models should be supported.

4. Real-world Use Case Examples

The requirement analysis above shows that generally information distribution should be better of as an infrastructure layer module, which provides to upper layer utilizations. In this section, we review some use cases from the real-world where an information distribution module with powerful functions do plays a critical role there.

4.1 Service-Based Architecture (SBA) in 3GPP 5G

In addition to Internet, the telecommunication network (i.e. carrier mobile wireless networks) is another world-wide networking system. The architecture of the upcoming 5G mobile networks from 3GPP has already been defined to follow a service-based architecture (SBA) where any network function (NF) can be dynamically associated with any other NF(s) when needed to compose a network service. Note that one NF can simultaneously associate with multiple other NFs, instead of being physically wired as in the previous generations of mobile networks. NFs communicate with each other over service-based interface (SBI), which is also standardized by 3GPP [3GPP.23.501].

In order to realize an SBA network system, detailed requirements are further defined to specify how NFs should interact with each other with information exchange over the SBI. We now list three requirements that are related to information distribution here.

1) NF Pub/Sub: Any NF should be able to expose its service status to the network and any NF should be able to subscribe the service status of an NF and get notified if the status is available. An concrete example is that a session management function (SMF) can subscribe the REGISTER notification from an access management function (AMF) if there is a new user entity trying to access the mobile network [3GPP.23.502].

2) Network Exposure Function (NEF): A particular network function that is required to manage the event exposure and distributions. In specific, SBA requires such a functionality to register network events from the other NFs (e.g. AMF, SMF and so on), classify the events and properly handle event distributions accordingly in terms of different criteria (e.g. priorities) [3GPP.23.502].

3) Network Repository Function (NRF): A particular network function where all service status information is stored for the whole network. An SBA network system requires all NFs to be stateless so as to improve the resilience as well as agility of

providing network services. Therefore, the information of the available NFs and the service status generated by those NFs will be globally stored in NRF as a repository of the system. This clearly implies storage capability that keeps the information in the network and provides those information when needed. A concrete example is that whenever a new NF comes up, it first of all registers itself at NRF with its profile. When a network service requires a certain NF, it first inquires NRF to retrieve the availability information and decides whether or not there is an available NF or a new NF must be instantiated [3GPP.23.502].

4.2 Vehicle-to-Everything

Carrier networks On-boarding services of vertical industries are also one of some blooming topics that are heavily discussed. Connected car is clearly one of the important scenarios interested in automotive manufacturers, carriers and vendors. 5G Automotive Alliance - an industry collaboration organization defines many promising use cases where services from car industry should be supported by the 5G mobile network. Here we list two examples as follows [5GAA.use.cases].

1) Software/Firmware Update: Car manufacturers expect that the software/firmware of their car products can be remotely updated/upgraded via 5G network in future, instead of onsite visiting their 4S stores/dealers offline as nowadays. This requires the network to provide a mechanism for vehicles to receive the latest software updates during a certain period of time. In order to run such a service for a car manufacturer, the network shall not be just like a network pipe anymore. Instead, information data have to be stored in the network, and delivered in a publishing/subscribing fashion. For example, the latest release of a software will be first distributed and stored at the access edges of the mobile network, after that, the updates can be pushed by the car manufacturer or pulled by the car owner as needed.

2) Real-time HD Maps: Autonomous driving clearly requires much finer details of road maps. Finer details not only include the details of just static road and streets, but also real-time information on the road as well as the driving area for both local urgent situations and intelligent driving scheduling. This asks for situational awareness at critical road segments in cases of changing road conditions. Clearly, a huge amount of traffic data that are real-time collected will have to be stored and shared across the network. This clearly requires the storage capability, data synchronization and event notifications in urgent cases from the network, which are still missing at the infrastructure layer.

4.3 Summary

Through the general analysis and the concrete examples from the real-world, we realize that the ways information are exchanged in the coming new scenarios are not just short and instant anymore. More advanced as well as diverse information distribution capabilities are required and should be generically supported from the infrastructure layer. Upper layer applications (e.g. ASAs in ANIMA) access and utilize such a unified mechanism for their own services.

5. Node Behaviors

In this section, we discuss how each autonomic node should behave in order to realize the information distribution module. In other words, we discuss the node requirement if an information distribution module is required across the ANI. Supporting the two communication models that may happen in the ANI necessarily involves node interactions and information data exchange. Specifically, we first introduce the node requirement for the instant communication model, and after that we introduce the node requirement for the asynchronous communication model.

5.1 Instant Information Distribution

In this case, sender(s) and receiver(s) are explicitly and immediately specified (e.g. the addresses of the receivers). Information will be directly distributed from the sender(s) to the receiver(s). This requires that every node is equipped by some signaling/transport protocols so that they can coordinate with each other and correctly deliver the information.

5.1.1 Instant P2P and Flooding Communications

We consider that current GRASP already provides some of the instant P2P and flooding communications capabilities.

Straightforwardly, it is natural to use the GRASP Synchronization message directly for P2P distribution. Furthermore, it is also natural to use the GRASP Flood Synchronization message for 1-to-all distribution.

However, as mentioned in Section 3, in some scenarios one node needs to actively send some information to another. GRASP Synchronization just lacks such capability. An un-solicited synchronization mechanism is needed. A relevant GRASP extension is defined in Section 6.

5.1.2 Instant Selective Flooding Communication

When doing selective flooding, the distributed information needs to contain the criteria for nodes to judge which interfaces should be sent the distributed information and which are not. Specifically, the criteria contain:

o Matching condition: a set of matching rules.

o Matching object: the object that the match condition would be applied to. For example, the matching object could be node itself or its neighbors.

o Action: what behavior the node needs to do when the matching object matches or failed the matching condition. For example, the action could be forwarding or discarding the distributed message.

The sender has to includes the criteria information in the message that carries the distributed information. The receiving node decides the action according to the criteria carried in the message. Still considering the criteria attached with the distributed information, the node behaviors can be:

o When the Matching Object is "Neighbors", then the node matches the relevant information of its neighbors to the Matching Condition. If the node finds one neighbor matches the Matching Condition, then it forwards the distributed message to the neighbor. If not, the node discards forwarding the message to the neighbor.

o When the Matching Object is the node itself, then the node matches the relevant inforshi mation of its own to the Matching Condition. If the node finds itself matches the Matching Condition, then it forwards the distributed message to its neighbors; if not, the node discards forwarding the message to the neighbors.

An example of selective flooding is briefly described in the Appendix A.

5.2 Asynchronous Information Distribution

Asynchronous information distribution happens in a different way where sender(s) and receiver(s) are normally not immediately specified. In other words, both the sender and the receiver may come up in an asynchronous way. First of all, this requires that the information can be stored; secondly, it requires an information publication and subscription (Pub/Sub) mechanism. (Corresponding protocol specification of Pub/Sub is defined in Section 6.)

Specifically, an information publisher 1) receives publishing requests from local AFs (also from ASAs), 2) decides where to store the published information, 3) updates corresponding event queues. On the other hand, an information subscriber registers its interests, 2) monitors event queues in the system and 3) trigger information retrieval if information of registered events are ready.

In general, each node requires two modules: 1) event queue (EQ) module and 2) information storage (IS) module shown in Figure. 1. These two modules should be integrated with the information distribution module. We introduce details of the two modules in the following sections.

```
+----------------------------------------+
| +----------------+   +---------------+ |
| |  Event Queue   |-|-| Info. Storage | |
| +----------------+   +---------------+ |
+----------------------------------------+
```

Figure 1. Components for asynchronous comm.

5.2.1 Event Queue

Event Queue (EQ) module is responsible for event classification, event prioritization and event matching.

Firstly, EQ module provides isolated event queues customized for different event groups. Specifically, two groups of AFs could have completely different purposes or interests, therefore EQ classification allows to create multiple message queues where only AFs interested in the same category of events will be aware of the corresponding event queue.

Secondly, events generated may have to be processed with different priorities. Some of them are more urgent than the normal and regular ones. Also between two event queues, their priorities may be different. EQ prioritization allows AFs to set different priorities on the information they published. Based on the priority settings in the event queue, matching and delivery of them will be adjusted. EQ module can provide several pre-defined priority levels for both intra-queue and inter-queue prioritizations.

Third, events in queues will be listened and if a publishing event is found and matched by a registration event, information retrieval will be triggered.

5.2.2 Information Storage

Events are closely related to the information. IS module handles how

to efficiently save and retrieve information for AFs across the network according to announced events. Any information that is published by AFs will be sent to the IS module, and the IS module decides where to store the information and how to index and retrieve it.

The IS module defines a syntax to index information, not only generating the hash index value (e.g. a key) for the information, but also mapping the hash index to a certain network node in ANI.

When data information is published by an AF (i.e. publishing events), it will be sent to the IS module. The IS module calculates its hash index (i.e. the key) and the location responsible for storing the information. The IS module confirms with the node chosen to store the information by negotiation. After that, if available, the IS module sends the information to there.

When data information has to be retrieved (i.e. subscribing events), a request from an AF will be also received by the IS module. IS module, by parsing the request, identifies the hash index of the information, which tells the location of the information as well. After that, the IS module requests the desired information and retrieves it once it is ready.

IS module can reuse distributed databases and key value stores like NoSQL, Cassandra, DHT technologies. storage and retrieval of information are all event-driven responsible by the EQ module.

5.2.3 Interface between IS and EQ Modules

EQ and IS modules are correlated. When an AF publishes information, not only an publishing event is translated and sent to EQ module, but also the information is indexed and stored simultaneously. Similarly, when an AF subscribes information, not only subscribing event is triggered and sent to EQ module, but also the information will be retrieved by IS module at the same time.

5.3 Summary

In summary, the general requirements for the information distribution module on each autonomic node are two sub-modules handling instant communications and asynchronous communications, respectively. For instant communications, node requirements are simple, in which signaling protocols have to be supported. With minimum efforts, reusing the existing GRASP is possible. For asynchronous communications, information distribution module requires event queue and information storage mechanism to be supported.

6. Protocol Specification (GRASP extension)

There are multiple ways to integrate the information distribution module. The principle we follow is to minimize modifications made to the current ANI.

We consider to use GRASP as an interface to access the information distribution module. The main reason is that the current version of GRASP is already an information distribution module for the cases of P2P and flooding. In the following discussions, we introduce how to complete the missing part.

6.1 Un-solicited Synchronization Message (A new GRASP Message)

In fragmentary CDDL, a Un-solicited Synchronization message follows the pattern:

```
unsolicited_synch-message = [M_UNSOLDSYNCH, session-id, objective]
```

A node MAY actively send a unicast Un-solicited Synchronization message with the Synchronization data, to another node. This MAY be sent to port GRASP_LISTEN_PORT at the destination address, which might be obtained by GRASP Discovery or other possible ways. The synchronization data are in the form of GRASP Option(s) for specific synchronization objective(s).

6.2 Selective Flooding Option

In fragmentary CDDL, the selective flood follows the pattern:

```
selective-flood-option = [O_SELECTIVE_FLOOD, +O_MATCH-CONDITION,
                          match-object, action]
O_MATCH-CONDITION = [O_MATCH-CONDITION, Obj1, match-rule, Obj2]
   Obj1 = text
   match-rule = GREATER / LESS / WITHIN / CONTAIN
   Obj2 = text
match-object = NEIGHBOR / SELF
action = FORWARD / DROP
```

The selective flood option encapsulates a match-condition option which represents the conditions regarding to continue or discontinue flood the current message. For the match-condition option, the Obj1 and Obj2 are to objects that need to be compared. For example, the Obj1 could be the role of the device and Obj2 could be "RSG". The match rules between the two objects could be greater, less than, within, or contain. The match-object represents of which Obj1 belongs to, it could be the device itself or the neighbor(s) intended to be flooded. The action means, when the match rule applies, the current

device just continues flood or discontinues.

6.3 Subscription Objective Option

In fragmentary CDDL, a Subscription Objective Option follows the pattern:

```
subscription-objection-option = [SUBSCRIPTION, 2, 2, subobj]
objective-name = SUBSCRIPTION
objective-flags = 2
loop-count = 2
subobj = text
```

This option MAY be included in GRASP M_Synchronization, when included, it means this message is for a subscription to a specific object.

6.4 Un_Subscription Objective Option

In fragmentary CDDL, a Un_Subscribe Objective Option follows the pattern:

```
Unsubscribe-objection-option = [UNSUBSCRIB, 2, 2, unsubobj]
objective-name = SUBSCRIPTION
objective-flags = 2
loop-count = 2
unsubobj = text
```

This option MAY be included in GRASP M_Synchronization, when included, it means this message is for a un-subscription to a specific object.

6.5 Publishing Objective Option

In fragmentary CDDL, a Publish Objective Option follows the pattern:

```
publish-objection-option = [PUBLISH, 2, 2, pubobj] objective-name
= PUBLISH
objective-flags = 2
loop-count = 2
pubobj = text
```

This option MAY be included in GRASP M_Synchronization, when included, it means this message is for a publish of a specific object data.

[Editor's Note]: Detailed node behavior and processing procedures of these new options will be introduced in the next version.

7. Security Considerations

The distribution source authentication could be done at multiple layers:

- Outer layer authentication: the GRASP communication is within ACP (Autonomic Control Plane, [I-D.ietf-anima-autonomic-control-plane]). This is the default GRASP behavior.

- Inner layer authentication: the GRASP communication might not be within a protected channel, then there should be embedded protection in distribution information itself. Public key infrastructure might be involved in this case.

8. IANA Considerations

TBD.

9. Normative References

[I-D.ietf-anima-grasp]
Bormann, D., Carpenter, B., and B. Liu, "A Generic Autonomic Signaling Protocol (GRASP)", draft-ietf-animagrasp-15 (Standard Track), October 2017.

10. Informative References

[RFC7575] Behringer, M., "Autonomic Networking: Definitions and Design Goals", RFC 7575, June 2015

[I-D.ietf-anima-autonomic-control-plane]
Eckert, T., Behringer, M., and S. Bjarnason, "An Autonomic Control Plane (ACP)", draft-behringer-anima-autonomic-control-plane-13, December 2017.

[I-D.ietf-anima-stable-connectivity-10]
Eckert, T., Behringer, M., "Using Autonomic Control Plane for Stable Connectivity of Network OAM", draft-ietf-anima-stable-connectivity-10, February 2018.

[I-D.ietf-anima-reference-model]
Behringer, M., Carpenter, B., Eckert, T., Ciavaglia, L., Pierre P., Liu, B., Nobre, J., and J. Strassner, "A Reference Model for Autonomic Networking", draft-ietf-anima-reference-model-05, October 2017.

[I-D.du-anima-an-intent]

Du, Z., Jiang, S., Nobre, J., Ciavaglia, L., and M. Behringer, "ANIMA Intent Policy and Format", draft-duanima-an-intent-05 (work in progress), February 2017.

[I-D.ietf-anima-grasp-api]
Carpenter, B., Liu, B., Wang, W., and X. Gong, "Generic Autonomic Signaling Protocol Application Program Interface (GRASP API)", draft-ietf-anima-grasp-api-00 (work in progress), December 2017.

[3GPP.23.501]
3GPP, "System Architecture for the 5G System", 3GPP TS 23.501 15.2.0, 6 2018, <http://www.3gpp.org/ftp/Specs/html-info/23501.htm>.

[3GPP.23.502]
3GPP, "Procedures for the 5G System", 3GPP TS 23.502 15.2.0, 6 2018, <http://www.3gpp.org/ftp/Specs/html-info/23502.htm>.

[5GAA.use.cases]
White Paper "Toward fully connected vehicles: Edge computing for advanced automotive communications", 5GAA <http://5gaa.org/news/toward-fully-connected-vehicles-edge-computing-for-advanced-automotive-communications/>

Appendix A.

GRASP includes flooding criteria together with the delivered information so that every node will process and act according to the criteria specified in the message. An example of extending GRASP with selective criteria can be:

- o Matching condition: "Device role=IPRAN_RSG"

- o Matching objective: "Neighbors"

- o Action: "Forward"

This example means: only distributing the information to the neighbors who are IPRAN_RSG.

Authors' Addresses

Bing Liu
Huawei Technologies
Q27, Huawei Campus
No.156 Beiqing Road
Hai-Dian District, Beijing 100095
P.R. China

Email: leo.liubing@huawei.com

Sheng Jiang
Huawei Technologies
Q27, Huawei Campus
No.156 Beiqing Road
Hai-Dian District, Beijing 100095
P.R. China

Email: jiangsheng@huawei.com

Xun Xiao
Munich Research Center
Huawei technologies
Riesstr. 25, 80992, Muenchen, Germany

Emails: xun.xiao@huawei.com

Artur Hecker
Munich Research Center
Huawei technologies
Riesstr. 25, 80992, Muenchen, Germany

Emails: artur.hecker@huawei.com

Zoran Despotovic
Munich Research Center
Huawei technologies
Riesstr. 25, 80992, Muenchen, Germany

Emails: zoran.despotovic@huawei.com