Information Distribution in Autonomic Networking
draft-liu-anima-grasp-distribution-05

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.

   These capabilities are fundamental and basic elements 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.

Status of this Memo

material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at
http://www.ietf.org/1id-abstracts.html

The list of Internet-Draft Shadow Directories can be accessed at
http://www.ietf.org/shadow.html

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 the communication models of information distribution into the following two:

> 1) An instant communication model where a sender builds a connection to send information (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 distribution function, utilized for stable connectivity as in [I-D.ietf-anima-stable-connectivity-10].

In this document, we summary possible scenarios of information distribution in autonomic networks (Section 2), and then discuss the technical requirements (Section 3) that an autonomic node has to fulfill. Moreover, possible ways to realize the information distribution module are presented.

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. Scenarios and Requirements of Information Distribution

We first summarize possible scenarios into the following categories. After that, we discuss the requirements of communication capabilities for the scenarios.

3.1 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.

3.2 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.

3.3 Requirements

Clearly, either the P2P scenario or the one-to-many scenario 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 combination of the 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 are needed.

[4](#). Node Requirements

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.

4.1 Requirements for 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.

4.1.1 Instant P2P and Flooding Communications

We consider that the GRASP in the existing ANI more or less already
can provide instant P2P and flooding communications with minimum
efforts.

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, because the Flood Synchronization behavior specified in
GRASP is identical to the the whole domain distribution scenario
described in Section 3.2.

However, as mentioned in Section 3.1, 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.

4.1.2 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 information 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.

4.2 Requirements for 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.

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.
```
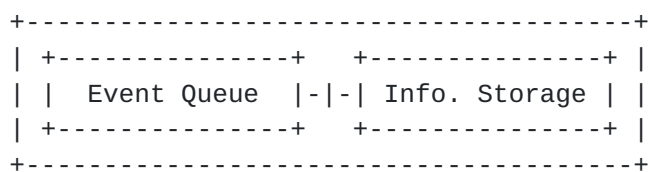
4.2.1 Event Queue

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.

4.2.2 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.

4.2.3 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.

4.3 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.

5. Integration with GRASP

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. What is missing are the support of 1) 1-to-Many
instant communications and 2) asynchronous communications. In the
following discussions, we introduce how to complete the missing part.

5.1 GRASP for instant communications

GRASP already supports instant communications for the cases of P2P
and flooding. In order to support 1-to-Many communication scenario,
where not all nodes in the network have to be the recipients, a
selective flooding is required.

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.

With selective flooding, GRASP covers the instant communication of
the information distribution module.

5.2 GRASP for asynchronous communications

For the asynchronous communication part, it is not covered by the
current GRASP. Trivially adding the Event Queue and Information
Storage modules spoils the current protocol design architecture of
GRASP. Our idea is to treat the asynchronous communication module as
a black box and design a reference point in GRASP, through which ASAs
could call the service of the black box.

The reference point mainly refers to a set of new messages that will
trigger asynchronous communications (e.g. Pub/Sub). We suggest that
the reference point contains the following messages:

   o Message for publishing information:

   o Message for subscribing information:

   o Message for unsubscribing information:

   o Message for unsubscribing information:

   o Message for even more:

Note that so far we do not specify the details of those potential new
messages for GRASP, to which more considerations are needed. Another
point is that the reference point suggested here can also refer to
some new objectives, instead of creating new messages.

In summary, with the selective flooding and the reference point of

accessing the asynchronous communication module, full communication
types can be fulfilled by the ANI. This provides a common interface
for the upper layer ASAs to communicate with each other for not only
control plane, but also data plane if needed.

6 Security Considerations

The distribution source authentication could be done at multiple
layers:

   o 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.

   o 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.

7 IANA Considerations

TBD.

8 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]
             Bormann, D., Carpenter, B., and B. Liu, "A Generic
             Autonomic Signaling Protocol (GRASP)", draft-ietf-
             animagrasp-15 (Standard Track), October 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.

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