Anima Bootstrapping for Network Management
draft-nmdt-anima-management-bootstrap-01

Abstract

This document points out the gaps of utilizing current Anima
technologies into a traditional centralized management network. It
raises some problems and requirements, based on which, as set of
solutions are proposed. (These solutions are called Anima
Bootstrapping for Network Management.)

Status of This Memo

This Internet-Draft is submitted in full conformance with the
provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering
Task Force (IETF). Note that other groups may also distribute
working documents as Internet-Drafts. The list of current Internet-
Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months
and may be updated, replaced, or obsoleted by other documents at any
time. It is inappropriate to use Internet-Drafts as reference
material or to cite them other than as "work in progress."

This Internet-Draft will expire on March 30, 2019.

Copyright Notice

Copyright (c) 2018 IETF Trust and the persons identified as the
document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal
Provisions Relating to IETF Documents
(https://trustee.ietf.org/license-info) in effect on the date of
publication of this document. Please review these documents
carefully, as they describe your rights and restrictions with respect
to this document. Code Components extracted from this document must
include Simplified BSD License text as described in Section 4.e of
1. Introduction

One typical usage of ANIMA technologies is that they serve as a stable management channel of the management systems, such as controllers or network management system (NMS) hosts. And these cases is also described in section 6 of [I-D.ietf-anima-autonomic-control-plane], with the purpose of management and controllability of ACP for the controllers or NMS hosts. However, in ANIMA networking, the autonomic nodes in ACP are self-configurable by default, most configuration of which is learning from neighboring nodes in decentralized ways. While in traditional networking, the configuration is got by the top-down ways from the centralized devices (such as controller, NMS hosts etc). These are the gaps and differences between the traditional networking and ANIMA networking.
Following this Introduction, Section 3 describes the problems of the integration of ACP and traditional centralized networking nodes, and then layout the solution requirements of it.

Based on the problems and solution requirements, this document discusses the Autonomic Structured Naming mechanism (in section Section 4), which provides meaningful names easy for human operation and maintenance; autonomous NMS/Controller discovery by the Autonomic Nodes Section 5; and topology discovery and collection Section 6 allowing the NMS/Controller to learn the topology of the managed network. Finally, discusses the capability of NMS/Controller correlating the naming and topology information to layout the whole picture of the managed entities in the Anima domain.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119] when they appear in ALL CAPS. When these words are not in ALL CAPS (such as "should" or "Should"), they have their usual English meanings, and are not to be interpreted as [RFC2119] key words.

This document uses the key words defined in [RFC7575].

The following additional terms are used throughout this document:

- AN: Autonomic Nodes.
- NMS: Networking Management System.
- EMS: Element Management System.
- NE: Networking Element.

3. Problems and Requirements

In ANIMA networking, every autonomic node has a global unique management address, this is the same with traditional networking. However, in traditional networking, the management addresses are globally planned by administrator. While in ANIMA networking, they are locally defined by the autonomic node itself using the information extracted from the domain certificate, called ULA addresses, as described in the section 5.8.2 of [I-D.ietf-anima-autonomic-control-plane]. In the view of centralized management tools, such as Networking Management System (NMS) hosts, there are usually two function modules included, the Element...
Management System (EMS) and the NMS core. The EMS is created by networking manager manually one by one for each networking element, using globally planned management addresses to establish SNMP sessions between EMS and networking elements. In ANIMA networking, because of the local definition of ULA addresses, it is difficult for networking managers to select which address to establish SNMP session or to do a correct functional deployment for each device.

To resolve the problems raised above, the requirements listed following must be satisfied:

- The autonomic nodes’ physical location and functional roles in networking MUST be initially set before running and can be dynamically discovered by the centralized management tools.

- The IP address of the centralized management tools MUST be published as service in the ANIMA networking, so that the autonomic nodes can trap the device information to the NMS host.

- By receiving the traps of the autonomic nodes, the centralized management tools must create the corresponding EMS in autonomic ways, not in manual ways by networking managers.

4. Autonomic Structured Naming

4.1. Requirements

- Representing each device

  - Inside a domain, each autonomic device needs a domain specific identifier.

- Uniqueness

  - The names MUST NOT collide within one autonomic domain.

  - It is acceptable that the names in different domains collide, since they could be distinguished by domains.

- Semantic Encoding

  - It is RECOMMENDED that the names encode some semantics rather than meaningless strings. This is for ease of management consideration that network administrators could easily recognize the device directly through the names.

- Consistency
The devices’ naming SHOULD follow the same pattern within a domain.

4.2. Name Format and Content

4.2.1. Structured Naming Format

- Naming Elements

The whole name string could be combined with several individual Naming Elements, each of which representing a specific semantic. For example: Location-DeviceType-FunctionalRole-DistinguisherNumber@NameofDomain.

The structure should be flexible that some elements are optional. When these optional fields are added, the name could still be recognized as the previous one.

- Element Attributes

Each Naming Element could have zero or more attributes describing detailed information of the element. The attributes do not need to be presented in the naming string, but be stored as metadata in the devices and be reported to the management system.

- Mandatory and Optional Naming Elements

In above example, the "DistinguisherNumber" and "NameofDomain" are mandatory whereas others are optional. At initial stage, the devices might be only capable of self-generating the mandatory fields and the "DeviceType" because of the lack of knowledge. Later, they might have learned the "Location" and "FunctionalRole" and added the fields into current name. However, the other devices could still recognize it according to the same "DistinguisherNumber".

4.2.2. Naming Content

The naming information SHOULD be suitable for the centralized tools to determine the location of the device and the functions to be deployed. The composing parts of the naming information are listed as following:

- Device Type

- Ownership
o Location. The physical location of the devices MUST be abbreviated and abstracted, and usually be setted into the device name fields of the naming information. How to abbreviate and abstract the location information, is a policy of the ISP and out-of-scope of this document.

o Role and Function. The roles and the functions to be deployed in the devices MUST be specified in high level words, and usually be setted into the device function description fields of the naming information. It MUST NOT include any detailed configuration parameters of the roles and functions. How to define the high level words of each function and role is out-of-scope of this document.

o TBD.

4.3. Autonomic Naming Approaches

4.3.1. Received and Self-generated Naming Elements

There are mainly two kinds of naming information, as the following.

- Received Naming Elements

The elements are advertised or injected by some external source. Operators are responsible for provisioning this kind of information. At least one of the interface types listed as following SHOULD be supported by the Autonomic Network:

- Hardware interface. The operator uses some out-of-bind tools to specify the naming information as a initial configure file, and write it to some storage material, such as USB devices, SD cards and etc. The physical interfaces MUST be supported by the devices to plug in the storage materials. In the system starting up procedure of the devices, it reads the naming information from the initial setted configure file, and reports the relation of the ULA addresses and device name to the centralized tools as described in the following sections of this document.

- Software interface. During the first startup of the device system, the operator uses some in-bind software interfaces (such as Command Line Interface (CLI), Web Brower and etc) to specify the naming information as a configure file, and to write it to its internal storage material, such as FLASH cards. If there is no naming information configure file, the starting procedure pauses and wait for the configuration of the naming information. After the configuration or if there is already an existing naming information file, the device continues the starting procedure,
reads out the naming information and reports the relation of the ULA addresses and device name to the management tools as described in the following sections of this document.

- Self-generated Naming Elements

The mandatory fields SHOULD be self-generated so that one device could name itself sufficiently without any advertised knowledges.

There should various methods for a device to extract/generate a proper word for each mandatory semantic fields (e.g. "DeviceType", "DistinguisherNum") from its self-knowledge.

4.3.2. Naming Metadata Storage

TBD.

5. Network Management Server/Controller Discovery

In order to connect to the centralized management tool, the AN devices MUST get acknowledgement of the address of it. In ANIMA networking, this MUST be done in autonomic ways. This section describes two methods for dynamic learning of the address of centralized management tools.

5.1. GRASP Method

This method is mandatory in ANIMA networking.

A centralized management tool is typically configured manually. When the centralized management tool joins an Autonomic Control Plane ([I-D.ietf-anima-autonomic-control-plane]) it MUST respond to GRASP ([I-D.ietf-anima-grasp]) M_NEG_SYN message. If the centralized management tool dose not take part in the ACP, the IPV6 address MUST be configured in one device (called Management Proxy) of ANIMA networking and that AN device MUST be responsible for responding to GRASP M_NEG_SYN message.

The discovery messages send from the AN devices to the centralized management tool (or Management Proxy) as follows:

discovery-message = [M_NEG_SYN, session-id, initiator, Centralized-tool-objective]
Centralized-tool-objective = ["AN_centralized_tool", F_SYNCH, loop-count, centralized-tool-address]
centralized-tool-address = ipv6-address

Figure 5: Centralized Management Tool Discovery
The value of centralized-tool-address field is zero. Other fields are followed the specification of GRASP.

The response from the Centralized Management Tool (or Management Proxy) will be a M_RESPONSE with the following parameters:

response-message = [M_RESPONSE, session-id, initiator, ttl, (+locator-option // divert-option), Centralized-tool-objective]

Figure 6: Centralized Management Tool Response

The value of centralized-tool-address field in Centralized-tool-objective is zero. Other fields are followed the specification of GRASP.

After the discovery procedure, the AN devices use M_REQ_SYN messages and the Centralized Management Tool (or Management Proxy) responds with M_SYNCH message as described in GRASP. In M_SYNCH message, the Centralized Management Tool (or Management Proxy) fills the centralized-tool-address field in Centralized-tool-objective of M_SYNCH message with the valid IPV6 address of Centralized Tool.

5.2. mDNS Method

This method is optional in ANIMA networking.

Performs DNS-based Service Discovery [RFC6763] over Multicast DNS [RFC6762] searching for the service 
"_centralize_management_address.udp.local". To prevent unacceptable levels of network traffic the congestion avoidance mechanisms specified in [RFC6762] section 7 MUST be followed. The AN devices SHOULD listen for an unsolicited broadcast response as described in [RFC6762]. This allows AN devices to avoid announcing their presence via mDNS broadcasts and instead silently join the centralized management tools by watching for periodic unsolicited broadcast res

6. Topology Discovery and Collection

6.1. Local Topoloty Discovery

For the traditional centralized tools such as NMS hosts, the Link Layer Discovery Protocol (LLDP) is used to discovery the neigboring nodes and the links between two nodes, this was specified in IEEE 802.1ab.
6.2. Topology Collection by NMS/Controller

GRASP is used to carry topology information to the NMS/Controller. (Details TBD.)

7. Device Names and Topology Mapping in the NMS/Controller

There are two information types for the AN devices that must be exchanged in ANIMA networking, so that the centralized management tools can get the acknowledgment of the topology of it. The fixed information, which is the name of the AN devices, and were initially setted by the operators in the setting up procedures as described in the previous sections. The dynamic information, which is autonomously created or learned by the AN devices themselves, including the ULA addresses of the ACP, domain name of the networking and etc.

8. Security

TBD.

9. IANA Considerations

TBD.

10. Acknowledgements

The main idea of this document was initiated by Gang Yan.

Valuable comments were received from Sheng Jiang etc.

This document was produced using the xml2rfc tool [RFC2629].

11. References

11.1. Normative References


11.2. Informative References

[I-D.behringer-anima-reference-model]

[I-D.ietf-anima-autonomic-control-plane]

[I-D.ietf-anima-grasp]


Authors' Addresses

Fanghong Duan
Huawei Technologies
N8, Huawei Campus
No. 101 Ruanjian Road
Yu-Hua-Tai District, Nanjing 210000
P.R. China

Email: duanfanghong@huawei.com