INTERNET-DRAFT
Intended Status: Standard Track
Expires: September 10, 2015

R. Huang
Huawei
Y. Yang
Yale University
March 9, 2015

Network Topology Service Framework for Carrier Network
draft-huang-opsawg-topology-service-framework-00

Abstract

This document introduces a distributed network topology service framework for operators to collect network topologies from the physical heterogeneous network, analyses and stores the topology information, and provides the path computing and topology information inquiring ability to applications (including network applications like OSS, and third-party applications).

Status of this Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

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

Network topology is a prerequisite for operators to carry many critical network management tasks, including resource managements, path computation, event correlation, fault monitoring and analysis. Current carrier networks are continually being refined and upgraded as needs change and technology evolves. Many technologies have developed protocol-specific ways to obtain network topologies for their own usages. For example, a router supporting OSPF maintains an identical area-topology database to determine the shortest path to any neighboring router; BGP maintains a consistent view of network topology to optimize routing and scale the network. However, when network topologies are required by applications, applications usually wish to be shielded from protocol-specifics information, even if network state information is collected in protocol-specific ways. It is obvious that none of these methods offer a general-purpose tool that can efficiently manage the network topology for a heterogeneous network with multiple technologies including BGP/OSPF/ISIS, and even SDN Open Flow, etc.

This document introduces a distributed network topology service framework for operators to collect network topologies from the physical heterogeneous network, analyses and stores the topology information, and provides flexible path computing and topology information inquiring ability to applications (including network applications like OSS, and third-party applications).

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].

This document uses the following terms:

BGP: Border Gateway Protocol

OSPF: Open Shortest Path First

IS-IS: Intermediate System to Intermediate System

SDN: Software Defined Network

OSS: Operational Support Systems

3. Network Topology Service Framework

This section describes the network topology service framework as

shown in Figure 1:

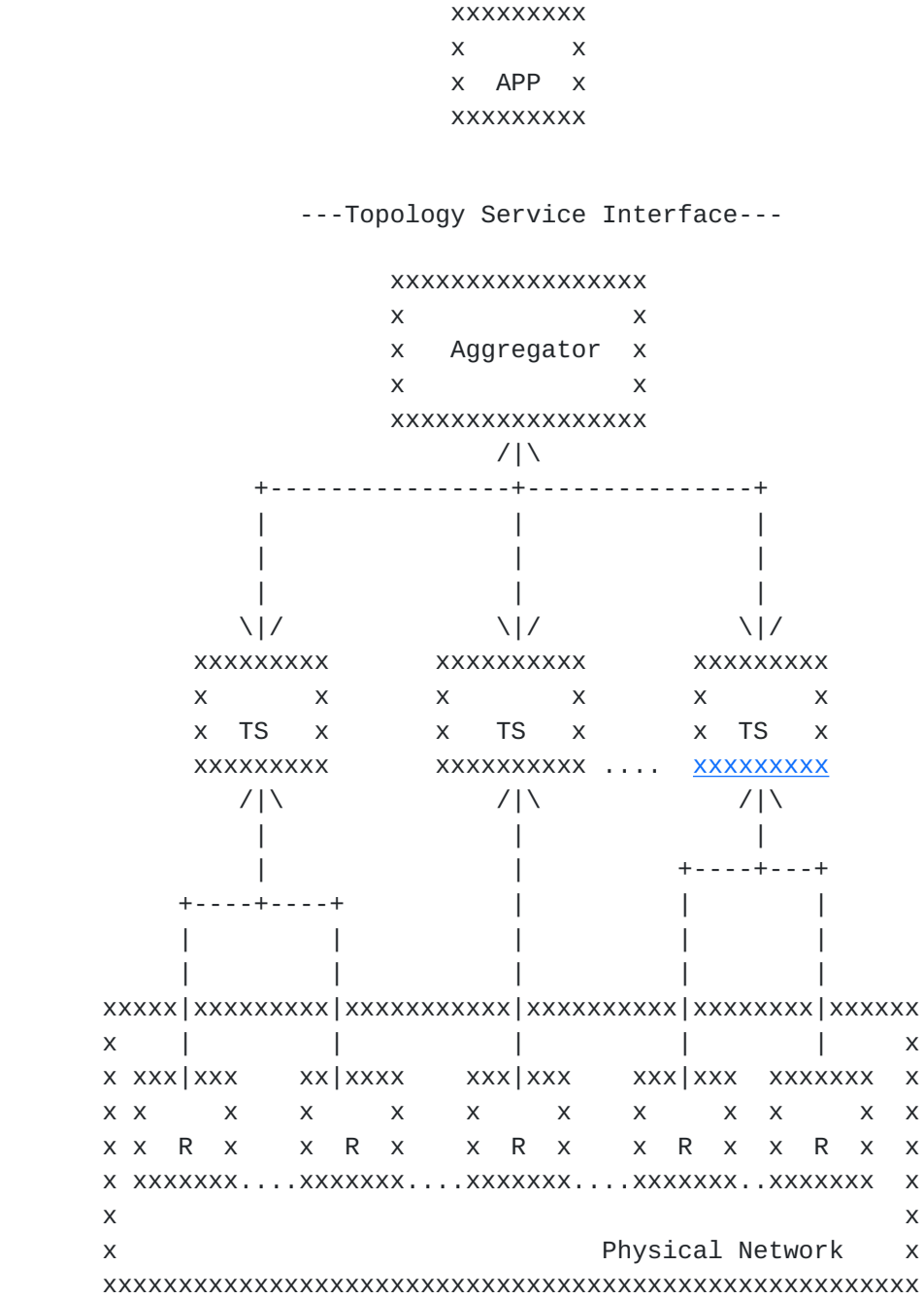

Figure 1: Framework of Network Topology Management System

The entities used in this framework are:

Topology Agent: A logical entity located in network devices like switches, routers, etc. It is responsible for reporting the topology information produced in some protocol-specific ways and network changes, event, or states to its topology server. One topology agent can only be controlled by one topology server to avoid global network topology duplicating.

Topology Server: A server that collects topology information from physical network for a subset of devices, analyses, abstracts, and stores the subset topology information in a protocol independent way. Usually, carrier's network is too large for a single topology server to handle. Thus, multiple topology servers are considered in this framework. Each of them is only responsible for a part of the global network. Topology server has the ability to calculate the most optimized path based on specific algorithms from applications. Different algorithms may lead to different results.

Aggregator: A server that maintains an abstract topology information among all topology servers. It does not perceive any detailed subset topology information as individual topology servers. This entity is only responsible for generating and maintaining relationship among different topology servers, and calculating the final optimized path based on the results calculated from some or all of the topology servers.

Application: It represents network applications like OSS, and third-party applications require to use network topology service.

The interfaces needed in this framework:

Interface between topology agent and topology server: An interface that can be used by TA to report different protocol-specific topology information, e.g., BGP/OSPF/IS-IS,or SDN OpenFlow, to TS. Besides, TA can use it to notify TS the changes, states, and events.

Interface between topology server and aggregator: Communication between topology servers and aggregator. It includes topology servers reporting their ingress and egress information to the aggregator, aggregator conveying applications' local topology algorithms information to topology servers, and topology servers returning their calculation results based on the local topology algorithms from applications to the aggregator.

Topology Service Interface: This interface is used by applications to communicate with the aggregator on path computation requesting

and abstract topology information obtaining. Applications use the interface to insert their own algorithms and requirements for the network topology service system to do some application specific calculations. Two kinds of algorithms are considered here: One for a topology server to calculate the most optimized path based on its subset topology information (local topology algorithm); The other for the aggregator to calculate the global and most optimized path based on the results from all topology servers and the abstract topology information generated by the aggregator (global topology algorithm).

4. Path Computing of Network Topology Service Framework

In SDN network, applications without knowledge of physical network can be benefit from the network topology management framework to obtain the most suitable and efficient network path based on which they can then do some programming.

The detailed steps of path computing is listed as following:

* TA discovers network topology and states/events.

* TA reports the protocol-specific network topology to TS.

* TS analyses the network topology information, and construct a generic subset network topology.

* TS reports its ingress and egress information to the aggregator.

* Aggregator generates an abstract topology information reflecting the relationship among all the TSs.

* Application inputs local topology algorithm, global topology algorithm, source information and destination information to the aggregator to request an optimized path.

* Aggregator instructs all of the TSs to calculate their own optimized path in their subset topologies based on the local topology algorithm of the application.

* TS reports its result to the aggregator.

* Aggregator calculate the final global optimized path based on the results of all the TSs, the abstract topology information, and the global topology algorithm.

When the number of TSs increasing, the performance of this framework may be reduced as all of the TSs need to do calculation. This can be

solved by applications inputting another algorithm or requirement to allow the aggregator filtering the relevant TSs before sending any instructions to TSs. Thus, only those TSs which are responsible for the network topology between the end-to-end network path are required to do the calculation. However, this function is optional here since some applications may need to all of the TSs to take part in the calculation.

5 Relationship with Other Existing IETF work

5.1 I2RS

I2RS is discussing a generic topology data model. However, current I2RS charter says it is not responsible to develop protocols, encoding languages, or data models. The topology work in I2RS can be considered to use in the interface between TA and TS. However, I2RS will not discuss a detailed topology service. The protocols and data models produced in I2RS can be considered in this work.

5.2 PCE

TBD.

6 Security Considerations

TBD.

7 IANA Considerations

This document does not require any IANA creations or modifications.

8 Acknowledgments

TBD.

9 References

9.1 Normative References

[KEYWORDS] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

9.2 Informative References

Authors' Addresses

Rachel Huang
Huawei
101 Software Avenue, Yuhua District
Nanjing 210012
China

EMail: rachel.huang@huawei.com

Y. Richard Yang
Yale University
51 Prospect St
New Haven, CT 06511
USA

EMail: yry@cs.yale.edu