TSVWG
Internet-Draft
Intended status: Informational
Expires: April 20, 2020

Y. Zhuang
B. Zhang
H. Pan
Huawei Technologies Co., Ltd.

October 18, 2019

Artificial Intelligence (AI) based ECN adaptive reconfiguration for datacenter networks
draft-zhuang-tsvwg-ai-ecn-for-dcn-00

Abstract

This document is to provide an artificial intelligence (AI) based ECN adaptive reconfiguration for datacenter networks.

Table of Contents

1. Introduction

1.1. Background

As defined in [RFC3168], Explicit Congestion Notification is introduced for IP to allow congestion to be signaled before dropping packets. As such, the latency of applications is reduced due to less retransmission of the dropped packets. Besides, MPLS also supports ECN defined in [RFC6679]. For tunneling, [RFC6040] defines how ECN should be constructed in the case of IP-in-IP tunnels.

Meanwhile, the upper layer transports protocols, like TCP in [RFC3168] and UDP based protocols DCCP in [RFC4341][RFC4342][RFC5632] and RTP in [RFC6679] are defined to support ECN-capable functions.

With ECN marking, active queue management (AQM) can choose a non-packet loss way to indicate congestion on the device, rather than dropping packets which might ask for packet retransmission and increase the latency. By using AQM in network devices, it can signal to common congestion-controlled transports to manage the queue length in the buffer and reduce the latency of traffics. Random Early

Detection (RED) specified in [RFC2309]is one of the AQM algorithms that recommended to be implemented in routers.

As stated in [RFC7567], with proper parameters, RED can be an effective algorithm. However, dynamically predicting the set of

parameters (minimum threshold and maximum threshold) is difficult. As a result, its present use in the Internet is limited. Other AQM algorithms have also been developed, while how to find proper parameters of algorithms for application traffics is still difficult and affect the network performance.

For data center networks, traffic patterns change with the deployment of applications like storage and high performance computing and changes of corresponding traffics which make the network more dynamic, while such applications have more restrict requirements on high throughput and ultra-low latency. In this area, a set of static ECN configurations suitable for all traffics at all time challenges.

With this, this document is to provide a way to seek ECN adaptive reconfiguration by using AI technologies in running data center network environment.

1.2. Intent

Our intent is to seek proper parameters of ECN adaptive reconfiguration by using artificial intelligence technologies to achieve self-tuning in a running data center network, so as to accommodate the changes of network resources to improve the network performance.

We also offer this as a starting point for seeking adaptive parameters for algorithms and network reconfigurations by using advanced technologies of AI. We do not change the way ECN works defined in [RFC3168]. With this, this document is to provide a way to achieve ECN adaptive reconfiguration by using AI technologies in dyanmic data center network environment.

1.3. 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 BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Architecture of the AI ECN datacenter networks

The following is a simple 2 layer data center network architecture with an analyzer to process the AI ECN adaptive reconfiguration with the changes of network traffics.

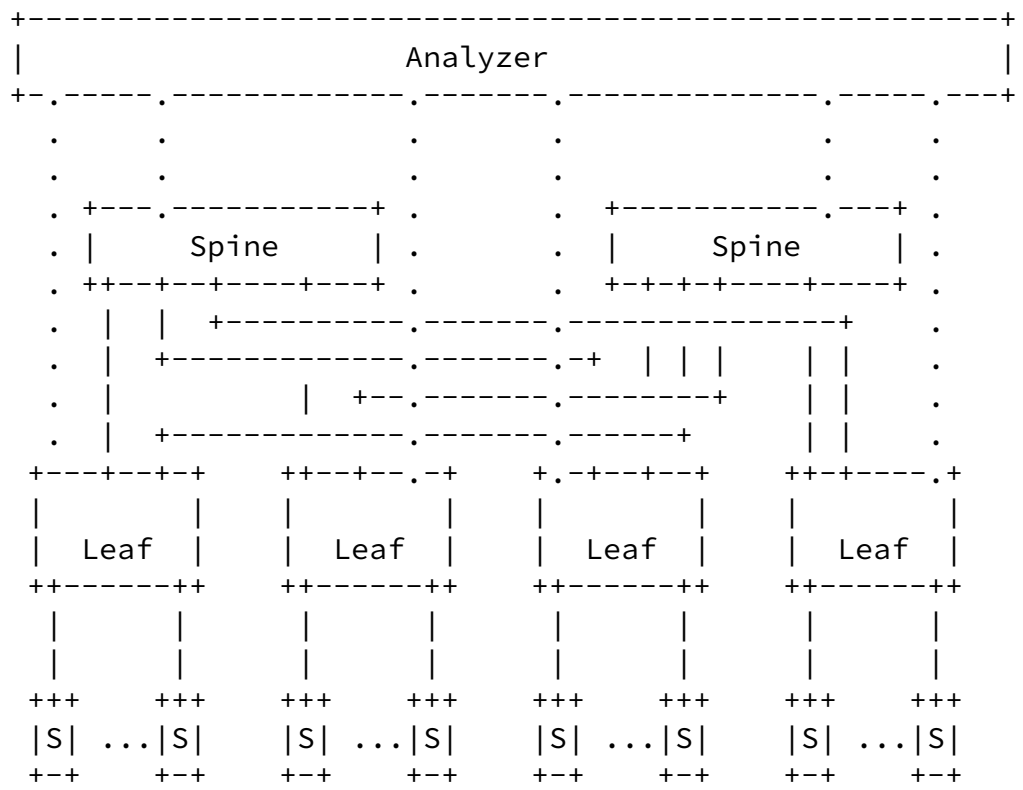

Figure 1. The architecture of a 2-layer data center network

The analyzer can be integrated with spine or can be an independent

device which is left for implementation. In this design, it is responsible for collecting device information and conducting the induction for proper parameters for ECN adaptive reconfiguration periodically.

3. Scene-based ECN adaptive reconfiguration with AI

The idea of AI ECN in this document is to identify the "scene" of the current network at some time based on the collected information over a period. The identified scene (which can also considered as a network traffic pattern)is one of the scenes that are collected and learned from datacenter networks running different traffics of various applications in training process. The ECN settings of these scenes are decided based on human experience. As such, the ECN parameters of current network can be tuned to the settings of the identified scene. This adaptive reconfiguration process is running periodically to accommodate changes of the running network environment due to traffic changes.

3.1. Scene Training

Scene training is the first process in the procedure. It composes of two steps. Firstly, construct typical scenes and generate a learning model to identify these scenes based on a set of network performance indicators. Secondly, provide proper ECN settings for these typical scenes based on human experience.

In the first step, it might need the network operator to select some typical applications and the combinations of traffics based on experience to be used as the typical training scenes. For these typical scenes, we run a learning algorithm (for example, neutral network) to learn the characteristics of these scenes from periodically collected network performance indicators.

The selected network performance indicators can be device's port bandwidth, queue size, etc al. which might be related to the applications and traffics in the networks.

While in the second step, human experience from network administrators can be used to provide proper ECN configurations for these typical scenes. AI technologies can also be used to enrich the

scene sets based on these human experience, which is left for implementation.

3.2. Scene Identification and ECN Adaptive Reconfiguration

In the practical network, the analyzer periodically collects information of selected network performance indicators from network nodes. The information is then used as input to the pre-learnt model and get the identified scene. The ECN settings of network devices will then be adaptively reconfigured to the parameters of the identified scene periodically.

The adaptive cycle of the period can be decided according to experience or it can be a training result in previous process defined in section 3.1.

4. Data collection and AI ECN adaptive reconfiguration

4.1. Data collection

In both training and adaptive reconfiguration process, the analyzer needs to collect information of the network i.e. a set of network performance indicators.

The data collection can be achieved by grpc or yang-push or other protocols.

4.2. ECN adaptive Reconfiguration

The adaptive reconfiguration of ECN in a running network environment can be achieved by control-plane protocols such as netconf.

5. Security Considerations

TBD

6. Manageability Consideration

TBD

7. IANA Considerations

No IANA action

8. References

8.1. Normative References

[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>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

8.2. Informative References

[RFC2309] Braden, B., Clark, D., Crowcroft, J., Davie, B., Deering, S., Estrin, D., Floyd, S., Jacobson, V., Minshall, G., Partridge, C., Peterson, L., Ramakrishnan, K., Shenker, S., Wroclawski, J., and L. Zhang, "Recommendations on Queue Management and Congestion Avoidance in the Internet", RFC 2309, DOI 10.17487/RFC2309, April 1998, <https://www.rfc-editor.org/info/rfc2309>.

[RFC3168] Ramakrishnan, K., Floyd, S., and D. Black, "The Addition of Explicit Congestion Notification (ECN) to IP", RFC 3168, DOI 10.17487/RFC3168, September 2001, <https://www.rfc-editor.org/info/rfc3168>.

[RFC4341] Floyd, S. and E. Kohler, "Profile for Datagram Congestion Control Protocol (DCCP) Congestion Control ID 2: TCP-like Congestion Control", RFC 4341, DOI 10.17487/RFC4341, March 2006, <https://www.rfc-editor.org/info/rfc4341>.

[RFC4342] Floyd, S., Kohler, E., and J. Padhye, "Profile for Datagram Congestion Control Protocol (DCCP) Congestion Control ID 3: TCP-Friendly Rate Control (TFRC)", RFC 4342,

DOI 10.17487/RFC4342, March 2006,
<https://www.rfc-editor.org/info/rfc4342>.

[RFC5632] Griffiths, C., Livingood, J., Popkin, L., Woundy, R., and Y. Yang, "Comcast's ISP Experiences in a Proactive Network Provider Participation for P2P (P4P) Technical Trial", RFC 5632, DOI 10.17487/RFC5632, September 2009, <https://www.rfc-editor.org/info/rfc5632>.

[RFC6040] Briscoe, B., "Tunnelling of Explicit Congestion Notification", RFC 6040, DOI 10.17487/RFC6040, November 2010, <https://www.rfc-editor.org/info/rfc6040>.

[RFC6679] Westerlund, M., Johansson, I., Perkins, C., O'Hanlon, P., and K. Carlberg, "Explicit Congestion Notification (ECN) for RTP over UDP", RFC 6679, DOI 10.17487/RFC6679, August 2012, <https://www.rfc-editor.org/info/rfc6679>.

[RFC7567] Baker, F., Ed. and G. Fairhurst, Ed., "IETF Recommendations Regarding Active Queue Management", BCP 197, RFC 7567, DOI 10.17487/RFC7567, July 2015, <https://www.rfc-editor.org/info/rfc7567>.

Acknowledgements

We would like to thank the following persons for their great efforts and contributions to the work: Huafeng Wen, Binghui Wu, Weiqin Kong, Ke Meng, Xitong Jia, Liang Shan, Siyu Yan, Weishan Deng, Boding Wang, Jungan Yan, Haonan Ye and Liang Zhang.

Authors' Addresses

Yan Zhuang
Huawei Technologies Co., Ltd.

Email: zhuangyan.zhuang@huawei.com

Bai Zhang

Huawei Technologies Co., Ltd.

Email: white.zhangbai@huawei.com

Haotao Pan
Huawei Technologies Co., Ltd.

Email: panhaotao@huawei.com