alto Z. Xing
Internet-Draft X. Di
Intended status: Informational H. Qi
Expires: 13 November 2022 Changchun University of Science and Technology
12 May 2022

The SDN-based MPTCP-aware and MPQUIC-aware Transmission Control Model using ALTO

draft-xing-alto-sdn-controller-aware-mptcp-mpquic-05

Abstract

This document aims to study and implement ~~MultiPath~~ Multipath Transmission Control Protocol (MPTCP) and MultipPath Q~~uick UDP Internet Connection~~UIC (MP-QUIC) using application layer traffic optimization (ALTO) in software defined network (SDN). In a software-defined network, ALTO server collects network cost indicators (including link delay, number of paths, availability, network traffic, bandwidth and packet loss rate etc.), and the controller extracts MPTCP or MPQUIC packet header to allocate MPTCP or MPQUIC packet to suitable transmission path according to the network cost indicators by ALTO, which can reduce the probability of transmission path congestion and improving path utilization.

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 13 November 2022.

Copyright Notice

Copyright (c) 2022 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 Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction

The ~~traditional~~ conventional TCP ~~protocol~~ only uses one path between ~~the~~ a server and ~~the~~ a client to ~~transmit~~ exchange data. In order to realize the simultaneous transmission of data ~~between~~ via multiple paths between ~~the~~a server and ~~the~~ a client, the ~~International Internet Engineering Task Force proposed and~~IETF standardized MPTCP [RFC6897] . MPTCP ~~realizes~~uses multiple paths between hosts to transmit data at the same time, but it is necessary to modify the operating system kernel to change the protocol stack of both parties in order to increase the MPTCP protocol. ~~Therefore,~~ MPTCP has disadvantages such as difficulty in deployment. In order to solve the drawbacks in the transmission network and adapt to the faster development of the Internet, Google proposed the HTTP/3 protocol which is ~~Quick UDP Internet Connection (~~QUIC) [RFC9000]. QUIC has many new features, such as: 0-RTT, forward error correction, connection migration, flexible congestion control, multiplexing without head-of-line blocking, easy deployment, and more. MPQUIC [MPQUIC] is a multi-path transmission protocol designed on the basis of QUIC. SDN [RFC7426] is a new network innovation architecture implemented by virtualization. By separating control and forwarding, it breaks the closedness of traditional network equipment, and uses programming to make network management more concise and efficient. flexible. ALTO [RFC7285] can obtain and ~~provide~~ expose global network information to ~~the~~ a controller, such as network traffic, link delay, etc. ~~The main multipath transmission protocols~~ MPTCP and MPQUIC have their own characteristics [MultipathTester]. The application of multipath transmission in SDN can greatly improve the transmission throughput.

Realize the coupling control of MPTCP and ~~mpquic~~ MP-QUCI subflows in the context of SDN ~~software defined network~~, obtain the network state information and allocate the optimal path according to the information conveyed in the ~~Alto~~ALTO Protocol, so as to improve the bandwidth utilization and resource allocation fairness, effectively alleviate the network congestion and realize the load balance between

Commenté [BMI1]: This one was experimental. It was obsoleted since then. Please update accordingly.

Commenté [BMI2]: A solution was proposed in RFC8803 to relax this.

Commenté [BMI3]: As per 9000, "QUIC is a name, not an acronym"

Commenté [BMI4]: TCP fast open/0-RTT TLS are also possible for TCP

Commenté [BMI5]: Is this inherent to the protocol or to the actors that are involved out there?

Commenté [BMI6]: You may also look at RFC7149

Commenté [BMI7]: SDN does not mandate nor preclude virtulaization.

Commenté [BMI8]: This separation was there since decades

Commenté [BMI9]: I would insist on the programmability

Commenté [BMI10]: As many may be deployed

Commenté [BMI11]: One of the scheduling policy that deployed is not to maximize the throughput, but to grab resources from a secondary link when needed. The policy should not overload some links (e.g., cellular links). I guess you have means to provide such a policy.

Also, there are users who have volume-based contracts. The use of multipath should not be aggressive as this will exhaust the quota, and lead to single-path. Please check https://www.rfc-editor.org/rfc/rfc8517#section-3.10 for more details paths.

At present, some scholars have studied the model of deploying MPTCP or MPQUIC in software-defined network, [QUICSDN] \ [SDN_for_MPTCP] \ [SDN_MPTCP], but their SDN controller cannot manage the headers of MPTCP and MPQUIC data packets at the same time, and cannot achieve unified management of MPTCP and MPQUIC links. The ALTO ~~protocol~~ Protocol can easily obtain various network states (including multiple SDNs, dynamic networks) from SDN without the internal details of the network provider, and deliver controller decisions [SDN_ALTO_proof] \ [SDN_ALTO], which is already a mature solution.

The purpose of this document is to:

Describe the model that ~~the~~ an SDN controller can use to extract MPTCP or MPQUIC data packets in the software-defined network.

According to the global information obtained by the AlTO, the controller allocates MPTCP or MPQUIC data packets with efficient transmission path.

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 [RFC2119].

3. Default transmission control mode of MPTCP or MPQUIC in SDN

In a software-defined network, the default controller cannot extract MPTCP or MPQUIC data packets. If MPTCP or MPQUIC are deployed and there are multiple transmission paths, the controller only selects one of the paths to transmit data, and the other paths are "idle" (i.e., there is only one path to transmit data). The utilization rate is low, and it is impossible to transmit data on multiple paths at the same time, resulting in low transmission efficiency.

4. Delivering functions by ALTO

A ~~Alto~~ ALTO server is used to obtain network status information, and ~~the~~an SDN controller ~~in SDN~~ is considered as ~~the~~ an ALTO client~~client of Alto~~. The ~~Alto~~ ALTO server~~ will~~ collects network cost indicators (including link delay, number of paths, availability, network traffic, bandwidth and packet loss rate).

5. Architectural Framework

The architectural framework of multi-path transmission model based on SDN controller MPTCP and MPQUIC using ALTO is shown in Figure 1.

```
+--------------Network Status Acquisition----------------+
|                     ALTO Server                        |
|        (Network topology, traffic distribution,        |
|               link delay/bandwidth)                    |
```

Commenté [BMI12]: Not sure I got what is meant here.

a mis en forme : Surlignage

a mis en forme : Surlignage

Commenté [BMI13]: Why it needs to extract these packets?

Commenté [BMI14]: How multipaths are used is policy-based. Where the default is defined?

Commenté [BMI15]: What is a “default controller” ?

Commenté [BMI16]: where?

a mis en forme : Surlignage

```
+---------------^---------------------------------------+
                |
                +------ALTO Protocol----+
                                        |
+-----------Control Plane-(ALTO Client)-v----------------+
|    +-------------------------------------------+       |
|    |     Extract MPTCP / MPQUIC header module  |       |
|    |          (Extract packet header)          |       |
|    +---------------------+---------------------+       |
|                          |                             |
|                     token or CID                       |
|                          |                             |
|    +---------------------v---------------------+       |
|    |             Path selection module         |       |
| +-->    (Select the appropriate path from      <--+    |
| |  |     the candidate paths - assigned path)    |  |    |
| |  +---------------------+---------------------+  |    |
| |                        |                   Allocated |
| |          +-----Allocate path------+          path    |
| |          |                        |           |      |
| |  +---------v----------+ +-----------v--------+  |    |
| |  |     Flow rules     | | Link management    |  |    |
| |  | generation module  | |       module       |  |    |
| |  |    (All switch     | |(Manage the mapping +--+    |
| |  |  assignment flow   | |table flows and save|       |
| |  |  tables for the    | |   the connection   |       |
| |  |  selected path)    | |    information)    |       |
| |  +---------+----------| +--------------------+       |
+-|------------|-----------------------------------------+
 Network       |
 status       +----Flow rules-----+
  |                               |
  |  +---------------Data Plane----v-------------+
  |  | +------------------+ +------------------+ |
  |  | |    SDN switch    | |    SDN switch    | |
  +--+ | (Forwarding flow | | (Forwarding flow | |
     | | rules and obtain | | rules and obtain | |
     | |  network status) | |  network status) | |
     | +------------------+ +------------------+ |
     +-------------------------------------------+
```

Figure 1 Schematic diagram of SDN-based MPTCP-aware and MPQUIC-aware transmission control model using ALTO

The SDN-based MPTCP and MPQUIC transmission control using ALTO model consists of three parts~~.~~:

* The first part is the network status acquisition module, which acquires basic network status information from ALTO.

* The second part is the control plane, that is the SDN controller, also the client of ALTO, which includes extracting MPTCP / MPQUIC header module, path selection module, flow rules generation module and link management module. The main function is to extract the header identifier token ~~or CID~~ of MPTCP (or CIDs ~~and~~ for MPQUIC) according to the data packet (For details, see Section 4), obtain the global information of the whole network according to AlTO and allocate suitable paths and put flow rules to switches according to the

Commenté [BMI17]: Both Source and Destination CID ?

Please note that multiple CIDs can be used even over the same path.

How these CIDs are correlated?

Who is generating the CID?

What are the entities that inject multipath signals.

global information of the entire network, and manage the links of the entire network at the same time.

* The third part is the data plane which is some OpenFlow switches. It executes the flow rules issued by the controller and realizes the forwarding of data packets.

6. Implementation and Deployment

```
        +---------------------+        +----------------------------+
        | Create a flow table |        |The packet p arrives at s1, |
        +----------+----------+        | and s1 performs flow rules <--+
                   |                   |   item matching on p       |  |
                   |                   +--------------+-------------+  |
        +----------v----------+                       |                |
        |Obtain Network Status|                       |                |
        |Extract packet header|<-----+                |                |
        +----------+----------+      |                v                |
                   |                 |               /\                |
       +-----------+------------+    |              /  \               |
       |           |            |    +---NO-----Match successful?      |
       v           v            v                   \  /               |
      /\          /\           /\                    \/                |
     /  \        /  \         /  \                   YES               |
  MP_CAPABLE     CID         MP_JOIN                  |                |
     \  /        \  /         \  /                    |                |
      \/          \/           \/        +------------v--------------+ |
      |           |            |        |Forward packet according to|-+
      |           |            v        |the flow rules instruction|
      |           |     +------+------+ +------------^--------------+
      |           |     |Extract token|              |
      |           |     +------+------+              |
      |           |            |                     |
      |           |            |                     |
      |     +------v----+ +-----v-------+             |
      |     | key=Q+CID | | key=T+token |             |
      |     +-----+-----+ +------+------+             |
      |           |              |                    |
      |           +------+-------+                    |
      |                  |                            |
      |                  v                            |
      |                 /\                            |
      |                /  \                           |
      |             Is there a key                    |
      |        +--in the flow table?--+               |
      |        |       \  /           |               |
      |       NO        \/           YES              |
      |        |                      |               |
      | +------v----------+   +-------v------+        |
      | |Extract source IP|   |              |        |
      | |destination IP   |   | Path of all  |        |
      | |source port      |   | subflows in  |        |
      | |destination port |   | value,RL     |        |
      | |and subflow      |   |              |        |
      | |identifier       |   |              |        |
      | +-------+---------+   +-------+------+        |
      |         |                     |               |
      |         |                     |               |
```

```
 | +-------v---------+   +-------v-------+        |
 | |Add the subflow  |   |Extract the    |        |
 | |meta information |   |subflow meta   |        |
 | |to value and then|   |information    |        |
 | |save <key:value> |   |and add it to  |        |
 | |to the flow rules|   |value          |        |
 | +-------+---------+   +-------+-------+        |
 +-------->|                     |                |
           |                     |                |
   +-------v---------+   +-------v------+         |
   |                 |   |Allocate a new|         |
   |Allocate the     |   |path to p, and|         |
   |first path to p  |   |route does not|         |
   |route            |   |belong to RL  |         |
   +-------+---------+   +-------+------+         |
           |                     |                |
           +----------+----------+                |
                      |                           |
                      |                           |
+---------------------v----------------------+    |
|Put forward and reverse flow rules to switch|----+
+--------------------------------------------+
```

Figure 2 The flow chart of the SDN-based MPTCP-aware and MPQUIC-aware multi-path transmission control model using ALTO

The flow chart of the SDN-based MPTCP-aware and MPQUIC-aware multi-path transmission control model using ALTO is shown in Figure 2. The transmission control model is realized by the following steps:

* Step 1. The ALTO server collects network cost indicators (including link delay, number of paths, availability, network traffic, bandwidth and packet loss rate), which are recorded as: G (V, e), network topology g, V is the vertex and E is the edge.

* Step 2. The SDN controller creates a mapping table flows for storing MPTCP or MPQUIC connection information, and each entry structure of the mapping table flows is <key:value>; wherein key is the unique identifier of MPTCP or MPQUIC connection, When the packet comes from MPTCP, key=T+token; and when the packet comes from MPQUIC, key=Q+CID (The letters T and Q are used to distinguish MPTCP and MPQUIC). value is a set of sub-stream meta-information, each item in the set is a sub-stream meta-information; each sub-stream meta-information consists of source IP, destination IP, source port, destination port, MPTCP (or MPQUIC) sub-stream identifier and the path route composition.

* Step 3. When the data packet p of a certain MPTCP or MPQUIC subflow reaches the first switch s1, the first switch s1 extracts the header field of the data packet p, extracts the source IP, source port, destination IP and the destination port matches the source IP, source port, destination IP and destination port of the flow table in the first switch s1 respectively, and judges whether the matching is successful. If so, go to step 13; if not, then the first switch s1 encapsulates the data packet p and forwards it to the SDN controller, and at the same time adds the data packet p to the waiting queue.

* Step 4. After receiving the data packet p, the SDN controller extracts the header field of the data packet p, extracts the

Commenté [BMI18]: What is this id for MPQUIC?

Commenté [BMI19]: More than one CID can be used. A CID can be withdrawn, etc.

Commenté [BMI20]: Which entity issues the data packet?

connection identifier of the data packet, and generates a key value, where when the data packet comes from MPTCP, key=T+token; When the packet comes from MPQUIC, key=Q+CID. Then query whether there is a key in the mapping table flows, if so, go to step 8, if not, go to step 5.

* Step 5. Extract the source IP, destination IP, source port, and destination port of the data packet p and generate a key value, where when the data packet comes from MPTCP, key=T+token; and when the data packet comes from MPQUIC, key=Q+CID .

* Step 6. ALTO to get basic network information. The controller calculates the threshold T according to the global network state information (network topology, number of switches, etc.). Using the depth-first traversal algorithm, find the available path set R={r_1,...,r_i,...,r_m } from all source nodes whose length does not exceed a certain threshold T to the destination node, r_i is the i available path, in the available path set Select a shortest path r_i in R as the path route of the sub-flow, where r_i=<s_(i,1),...,s_(i,j),...>, s_(i,j) represents the i available path The switch numbered j, where i belong to [1,m],j belong to [1,T].

* Step 7. Use the MPTCP and MPQUIC connection identifiers as the unique identifier key of the MPTCP and MPQUIC connections, where the key is the unique identifier of the MPTCP and MPQUIC connections. When the data packet comes from MPTCP, key=T+token; and the data packet comes from In MPQUIC, key=Q+CID. The source IP, source port, destination IP, destination port, MPTCP, MPQUIC sub-flow identifier and path route of the data packet p are added to the set value of sub-flow meta information as sub-flow meta-information, and then the <key:value> The form is saved to the mapping table flows, and go to step 11.

* Step 8. The SDN controller updates the flows table according to the global information of the network, and takes out the value from the connection identifier, and then composes all paths in the value into a set RL={r_1,r_2,...}.

* Step 9. The SDN controller searches for a suitable disjoint path for the data packet p according to the method in Step 5, and sets the found path as route=r_i, where r_i not belong to RL.

* Step 10. Extract the source IP, destination IP, source port, destination port, and MPTCP, MPQUIC sub-flow identifiers of the data packet p, and convert the source IP, source port, destination IP, destination port, MPTCP (or MPQUIC) sub-flow identifiers and the path route is added to the value as sub-flow meta information.

* Step 11. The SDN controller uses the source IP, source port, destination IP and destination port to issue the flow table to all switches in the route route, and set the route route=r_i=<s_(i,1),...,s_(i,j-1),s_(i.j),s_(i,j+1),...>, for the switch s_(i,j), the flow entry sent is the source IP, source port to the destination, the data packets of IP and destination port are forwarded to s_(i,j+1).

* Step 12. The controller sends the reverse flow table to all

switches on the route route and sets the route route=r_i=<s_(i,1),...,s_(i,j-1),s_(i,j),s_(i,j+1),...>, for the switch s_(i,j) ,the flow table entry sent is to forward the data packets from the destination IP, destination port to source IP, and source port to s_(i,j-1).

* Step 13. The switch already contains a flow entry for processing the data packet p, and forwards the data packet according to the rules defined by the flow entry, and completes the processing of the data packet p. Step 3 is executed when the forwarding fails or the processing of other subsequent data packets returns.

7. Security Considerations

The transmission control model uses the default security mechanism of SDN\ALTO\MPTCP\QUIC in the network, and does not modify the default security mechanisms such as encryption and authentication models [RFC7426], [RFC7285], [RFC6824] and [RFC9000].

8. IANA Considerations

TBD.

9. Discussion

The SDN transmission control model proposed in this document can simultaneously identify MPTCP and MPQUIC data packets and allocate optimal paths according to the network status obtained by ALTO, which expands the application scope of MPTCP and MPQUIC. In order to verify its comprehensive transmission performance, a fat-tree data center network is designed. The transmission control method proposed in this document improves the throughput by about 3 times compared to the default transmission control method. This model also supports data transmission in multiple software-defined networks, and can also be applied to satellite networks, marine networks, etc. to transmit data.

10. Acknowledgments

The authors thank all reviewers for their comments.

11. References

11.1. 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>.

[RFC6824] Ford, A., Raiciu, C., Handley, M., and O. Bonaventure, "TCP Extensions for Multipath Operation with Multiple Addresses", RFC 6824, DOI 10.17487/RFC6824, January 2013, <https://www.rfc-editor.org/info/rfc6824>.

[RFC6897] Scharf, M. and A. Ford, "Multipath TCP (MPTCP) Application Interface Considerations", RFC 6897, DOI 10.17487/RFC6897, March 2013, <https://www.rfc-editor.org/info/rfc6897>.

[RFC7285] Alimi, R., Ed., Penno, R., Ed., Yang, Y., Ed., Kiesel, S., Previdi, S., Roome, W., Shalunov, S., and R. Woundy, "Application-Layer Traffic Optimization (ALTO) Protocol", RFC 7285, DOI 10.17487/RFC7285, September 2014, <https://www.rfc-editor.org/info/rfc7285>.

[RFC7426] Haleplidis, E., Ed., Pentikousis, K., Ed., Denazis, S., Hadi Salim, J., Meyer, D., and O. Koufopavlou, "Software-Defined Networking (SDN): Layers and Architecture Terminology", RFC 7426, DOI 10.17487/RFC7426, January 2015, <https://www.rfc-editor.org/info/rfc7426>.

[RFC9000] Iyengar, J., Ed. and M. Thomson, Ed., "QUIC: A UDP-Based Multiplexed and Secure Transport", RFC 9000, DOI 10.17487/RFC9000, May 2021, <https://www.rfc-editor.org/info/rfc9000>.

11.2. Informative References

[MPQUIC] "Multipath Extension for QUIC", <https://www.ietf.org/archive/id/draft-lmbdhk-quic-multipath-00.html>.

[MultipathTester] "Coninck Q D , Bonaventure O . MultipathTester: Comparing MPTCP and MPQUIC in Mobile Environments[C]// 2019 Network Traffic Measurement and Analysis Conference (TMA). 2019.", <https://10.23919/TMA.2019.8784653>.

[QUICSDN] "Kumar P , Chen J , Dezfouli B . QuicSDN: Transitioning from TCP to QUIC for Southbound Communication in SDNs[J]. 2021.", <https://ui.adsabs.harvard.edu/abs/2021arXiv210708336K>.

[SDN_ALTO] "V. K. Gurbani, M. Scharf, T. V. Lakshman, V. Hilt and E. Marocco, "Abstracting network state in Software Defined Networks (SDN) for rendezvous services," 2012 IEEE International Conference on Communications (ICC), 2012, pp. 6627-6632.", <https://doi.org/10.1109/ICC.2012.6364858>.

[SDN_ALTO_proof] "Faigl, Z. , Z. Szabo , and R. Schulcz . "Application-layer traffic optimization in software-defined mobile networks: A proof-of-concept implementation." IEEE(2014):1-6.", <https://doi.org/10.1109/NETWKS.2014.6959200>.

[SDN_for_MPTCP] "Hussein A , Elhajj I H , Chehab A , et al. SDN for MPTCP: An enhanced architecture for large data transfers in datacenters[C]// IEEE International Conference on Communications. IEEE, 2017.", <https://doi.org/10.1109/ICC.2017.7996653>.

[SDN_MPTCP] "7. K. Gao, C. Xu, J. Qin, S. Yang, L. Zhong and G. Muntean, "QoS-driven Path Selection for MPTCP: A Scalable

SDN-assisted Approach," 2019 IEEE Wireless Communications
and Networking Conference (WCNC), 2019, pp. 1-6,",
<https://doi.org/10.1109/WCNC.2019.8885585>.

Authors' Addresses

Ziyang Xing
Changchun University of Science and Technology
Changchun
Email: more60@163.com

Xiaoqiang Di
Changchun University of Science and Technology
Changchun
Email: dixiaoqiang@cust.edu.cn

Hui Qi
Changchun University of Science and Technology
Changchun
Email: qihui@cust.edu.cn