Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: May 1, 2020

H. Li
Q. Wu
B. Wu
Q. Zhang
J. Zhou
J. Liu
Tsinghua University

October 29, 2019

Trusted Multipath-TCP (MPTCP) extension
draft-hewu-mptcp-trust-00

Abstract

Multipath TCP (MPTCP) adds the capability of using multiple paths to a regular TCP session and is being deployed extensively. Source Address Validation (SAV) technologies are proposed to prevent network nodes from spoofing others' IP addresses and thus improve the accountability of networks. This document proposes a trusted MPTCP extension based on SAV, which enables MPTCP to work with SAV and thus improve the accountability of MPTCP connections. This extension doesn't intend to replace the security solutions to resolving IP forged attacks, like Hash-based Message Authentication Code (HMAC), but to improve the accountability of them and the whole connection.

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 May 1, 2020.

Table of Contents

1. Introduction

Multipath TCP (MPTCP) [I-D.ietf-mptcp-rfc6824bis][RFC6824] adds the capability of using multiple paths to a regular TCP session and is being deployed extensively. The main threats of MPTCP are described in [RFC6181], [RFC7430] and they are mainly caused by forged control packets sent by malicious hosts with forged IP addresses. Source Address Validation (SAV) methods like Source Address Validation Architecture (SAVA) [RFC5210] and Source Address Validation Improvement (SAVI) [RFC7039] are developed to prevent nodes from spoofing others' IP addresses with finer-grained ingress filtering.

This document proposes a SAV based MPTCP enhancement, which enables MPTCP to work with SAV and thus improve the accountability of MPTCP connections.

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].

This document uses the MPTCP terminology introduced in [I-D.ietf-mptcp-rfc6824bis] [RFC6824].

SAV: Source Address Validation.

SAVA: Source Address Validation Architecture, refer to [RFC5210].

SAVI: Source Address Validation Improvement, refer to [RFC7039].

Trusted Connection: A TCP connection which SAV is deployed on both hosts.

HMAC: Hash-based Message Authentication Code, refer to [RFC2104].

Trusted Address: IP address protected by SAV which means it can't be forged.

3. Operation Overview

This section specifies the behavior of source address validation based MPTCP enhancement.

Note that this enhancement doesn't intend to replace the security solutions to resolving IP forged attacks, like Hash-based Message Authentication Code (HMAC), but to improve the accountability of them and the whole connection. However, security-oriented operations which protect control packets from being forged like HMAC MAY be omitted if there is a trusted connection between the both hosts, for the accountability of the connection is guaranteed by SAV. Other packets and their processing which are not mentioned in this document stay the same as related description in [I-D.ietf-mptcp-rfc6824bis][RFC6824].

3.1. Trusted Address notification

A Trust Flag is set in ADD_ADDR option to indicate the IP address is trusted or not. After Host B receives an ADD_ADDR option, it MUST add the binding entry to Trusted Path Binding Table (TPBT, see section 6) and send a packet to Host A to indicate it has successfully received ADD_ADDR option.

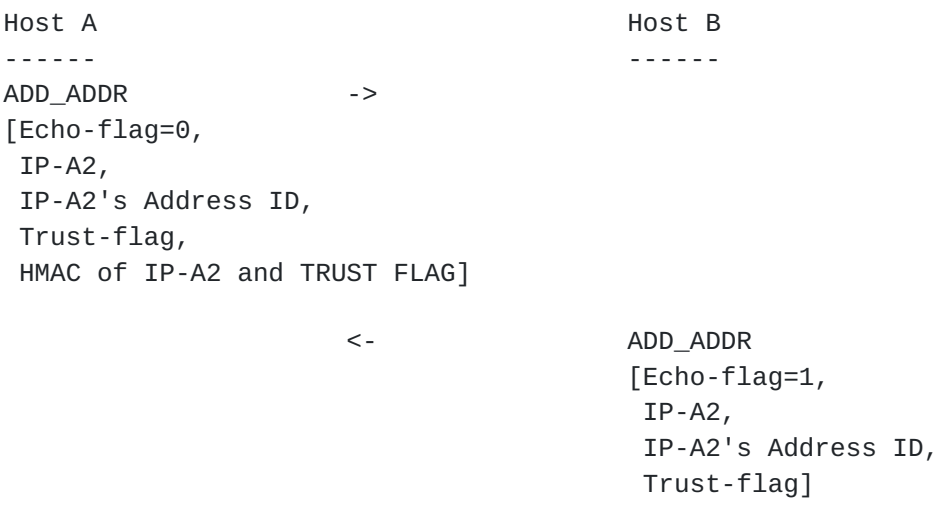

Figure 1

HMAC-A = HMAC(Key=(Key-A+Key-B), Msg=(IP-A2+TRUST Flag))

Note that if the ADD_ADDR option is transmitted through a trusted connection, the HMAC-A MAY be omitted. If the HMAC is transmitted and it's incorrect, the ADD_ADDR packet MUST be silently discarded.

3.2. Trusted Connection notification

When a MPTCP subflow or an initial MPTCP flow is established, the trust flag of this flow MAY not be known by both hosts. Examples as follows:

(1) There is no ADD_ADDR option sent by each other before the initial MPTCP flow is established.

(2) Host B starts initializing a subflow after receiving the ADD_ADDR option of Host A without sending its own ADD_ADDR option, when the subflow is established, Host A does not know the trust flag about this subflow if the Host A's address of this subflow is trusted.

An ADDR_TRUST option is proposed to notify hosts the trust flag. After a subflow is established, if the host does not know the trust flag of this subflow, it will add an entry (Trust=False) with peer address to the TPBT, and send an ADDR_TRUST option to the peer to ask for the trust flag of the peer's corresponding address. Once a host receives a ADDR_TRUST (E=0) packet and the HMAC is correct, it adds an entry(Trust=True) to the TPBT according the packet if it has not ever received an ADD_ADDR packet from this address. Also, the host checks the trust flag of the local address of the connection corresponding to the peer address: if the address is trusted, the host will send a ADDR_TRUST(E=1) packet with local address and HMAC

of two address, otherwise it does nothing. The host sent the ADDR_TRUST(E=0) packet will set the corresponding trust flag to True if it receives the ADDR_TRUST(E=1) packet.

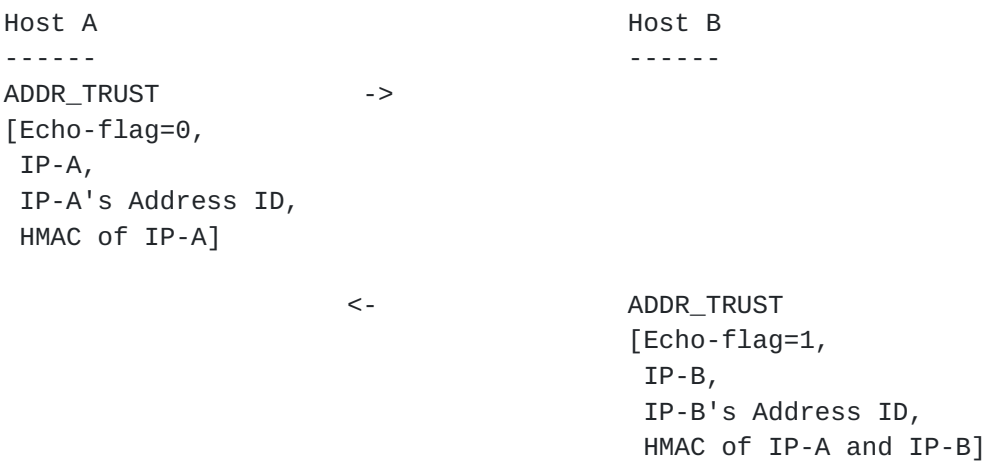

Figure 2

HMAC-A = HMAC(Key=(Key-A+Key-B), Msg=(IP-A))

HMAC-B = HMAC(Key=(Key-A+Key-B), Msg=(IP-A+IP-B))

Note that if the ADDR_TRUST option is transmitted through a trusted connection, the HMAC-A and HMAC-B MAY be omitted. If the HMAC is transmitted and it's incorrect, the ADDR_TRUST packet MUST be silently discarded.

3.3. Control packets processing

Before sending a control packet, the sender MUST check the TPBT: if there is a Trusted Connection whose source address and destination address are both trusted, it sends the control packet via the Trusted Connection and other security-oriented operations are OPTIONAL; if there is no Trusted Connection, the processing is the same as [I-D.ietf-mptcp-rfc6824bis][RFC6824].

4. ADD_ADDR extension

The ADD_ADDR option on packets includes 4 bits of flags, 2 of which are currently reserved and MUST be set to zero by the sender. The third bit, labeled "T", indicates the IP address is trusted(T=1) or not(T=0). The final bit, labeled "E", is used to Guarantee the reliability: a receiver receiving a fresh ADD_ADDR option (where E=0), will send the same option back to the sender, but not including the HMAC, and with E=1.

The format of ADD_ADDR option is shown in Figure 3.

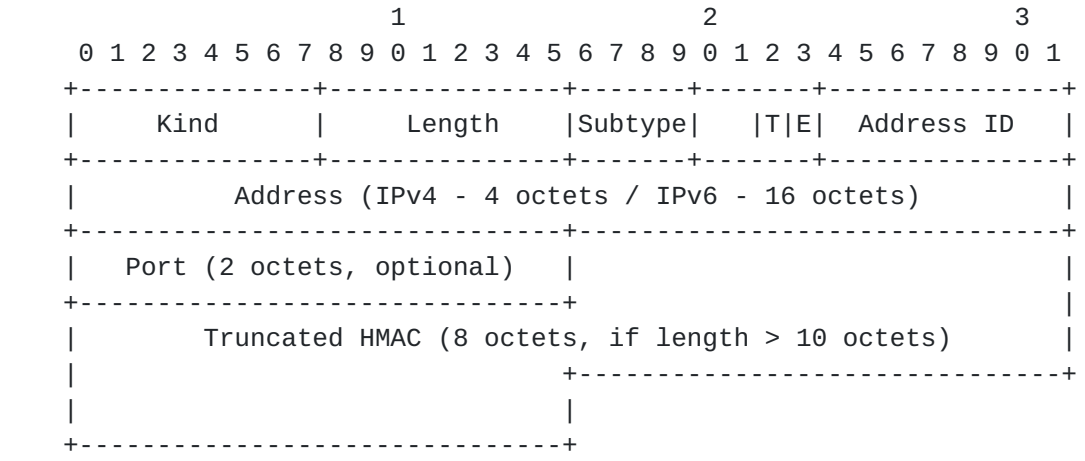

Figure 3: Add Address (ADD_ADDR) Option with HMAC

5. ADDR_TRUST option

The ADDR_TRUST option on packets includes 4 bits of flags, 3 of which are currently reserved and MUST be set to zero by the sender. The final bit, labeled "E", used to guarantee the reliability, a receiver receiving a fresh ADDR_TRUST option (where E=0), will send the same option back to the sender, but with HMAC of both Address, and with E=1.

The format of ADDR_TRUST option is shown in Figure 4.

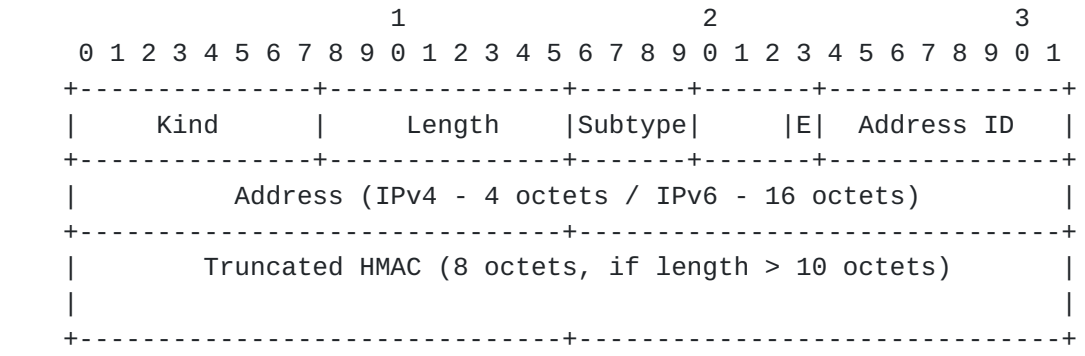

Figure 4: Address Trust (ADDR_TRUST) Option with HMAC

6. Trusted Path Binding Table (TPBT)

The Trusted Path Binding Table, which is implemented at the terminal, is used to contain the bindings between the available sockets of peer and their trust flags. This table uses "SubFlow" as the primary key which contains a "Sip" meaning the source IP address of the subflow

and a "Dip" for the destination IP address. Each entry in TPBT contains a "SipTrust" field representing whether the "Sip" is trusted and a "DipTrust" field for "Dip"; a Lifetime field is used to save the state of the life cycle and when the life cycle expires, the corresponding entry will be deleted; in addition, an Other field that is used to store other information or further extensions in the future.

The following table is an example of TPBT.

SubFlow	SipTrust	DipTrust	Lifetime	Other
Sf(Sip1,Dip1)	True	True	65535	/
Sf(Sip1,Dip2)	True	False	10000	/
Sf(Sip2,Dip1)	False	True	10000	/
Sf(Sip2,Dip2)	False	False	0	/

Table 1: An Example of TPBT

7. Acknowledgements

8. IANA Considerations

This memo includes no request to IANA.

9. Security Considerations

Some considerations on man-in-the-middle attacks may be raised for this extension. SAV methods like SAVI and SAVA are proposed to improve network accountability and thus defend attacks including man-in-the-middle attacks. This document is proposed to allow MPTCP work with SAV, so man-in-the-middle will not be a problem if SAV is deployed extensively.

10. Informative References

[I-D.ietf-mptcp-rfc6824bis]
Ford, A., Raiciu, C., Handley, M., Bonaventure, O., and C. Paasch, "TCP Extensions for Multipath Operation with Multiple Addresses", June 2019, <https://datatracker.ietf.org/doc/html/draft-ietf-mptcp-rfc6824bis-18>.

[RFC2104] Krawczyk, H., Bellare, M., and R. Canetti, "HMAC: Keyed-Hashing for Message Authentication", RFC 2104, DOI 10.17487/RFC2104, February 1997, <https://www.rfc-editor.org/info/rfc2104>.

[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>.

[RFC5210] Wu, J., Bi, J., Li, X., Ren, G., Xu, K., and M. Williams, "A Source Address Validation Architecture (SAVA) Testbed and Deployment Experience", RFC 5210, DOI 10.17487/RFC5210, June 2008, <https://www.rfc-editor.org/info/rfc5210>.

[RFC6181] Bagnulo, M., "Threat Analysis for TCP Extensions for Multipath Operation with Multiple Addresses", RFC 6181, DOI 10.17487/RFC6181, March 2011, <https://www.rfc-editor.org/info/rfc6181>.

[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>.

[RFC7039] Wu, J., Bi, J., Bagnulo, M., Baker, F., and C. Vogt, Ed., "Source Address Validation Improvement (SAVI) Framework", RFC 7039, DOI 10.17487/RFC7039, October 2013, <https://www.rfc-editor.org/info/rfc7039>.

[RFC7430] Bagnulo, M., Paasch, C., Gont, F., Bonaventure, O., and C. Raiciu, "Analysis of Residual Threats and Possible Fixes for Multipath TCP (MPTCP)", RFC 7430, DOI 10.17487/RFC7430, July 2015, <https://www.rfc-editor.org/info/rfc7430>.

Authors' Addresses

Hewu Li
Tsinghua University
Institute for Network Sciences and Cyberspace, Tsinghua University
Beijing 100084
China

Email: lihewu@cernet.edu.cn

Qian Wu
Tsinghua University
Institute for Network Sciences and Cyberspace, Tsinghua University
Beijing 100084
China

Email: wuqian@cernet.edu.cn

Boyang Wu
Tsinghua University
Institute for Network Sciences and Cyberspace, Tsinghua University
Beijing 100084
China

Email: wuboyangyawn@hotmail.com

Qi Zhang
Tsinghua University
Institute for Network Sciences and Cyberspace, Tsinghua University
Beijing 100084
China

Email: qi-zhang19@mails.tsinghua.edu.cn

Jiang Zhou
Tsinghua University
Institute for Network Sciences and Cyberspace, Tsinghua University
Beijing 100084
China

Email: zhou-j17@mails.tsinghua.edu.cn

Jun Liu
Tsinghua University
Institute for Network Sciences and Cyberspace, Tsinghua University
Beijing 100084
China

Email: juneliu@tsinghua.edu.cn