Internet Engineering Task Force
Internet-Draft
Intended status: Standards Track
Expires: 8 August 2026

B. Liang
Tsinghua University
Y. Xiang
Yunshan Networks
X. Shi
X. Yin
Tsinghua University

4 February 2026

TCP Provenance Identifier Option
draft-liang-tcp-provenance-option-01

Abstract

This document describes a TCP option that carries a Provenance Identifier (ProvID) to enable correlation of TCP connections when transport-layer identifiers change along the path.

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 8 August 2026.

Copyright Notice

Table of Contents

1. Introduction

In administrative domains (e.g., cloud platforms, enterprise networks, and data centers), TCP traffic often traverses devices such as NATs, load balancers, and service proxies that rewrite transport-layer identifiers or terminate and re-originate TCP connections. As a result, a single end-to-end exchange between two workloads may correspond to a sequence of distinct TCP connections within the domain. This document refers to that end-to-end exchange as a "logical communication".

These transformations break provenance continuity. Observations of TCP traffic at different points in the domain cannot reliably be associated with the same logical communication, and operators cannot determine which workload instance originally initiated a connection once rewriting has occurred.

This document defines an experimental TCP option that carries a small Provenance Identifier (ProvID). A ProvID is a compact value generated for the duration of a logical communication using workload-scoped attributes, such as host IP with process identifier. As illustrated in Figure 1, the ProvID enables provenance correlation across rewriting boundaries within the domain.

The ProvID option is intended for use within administrative domains and is not designed for use on the open Internet.

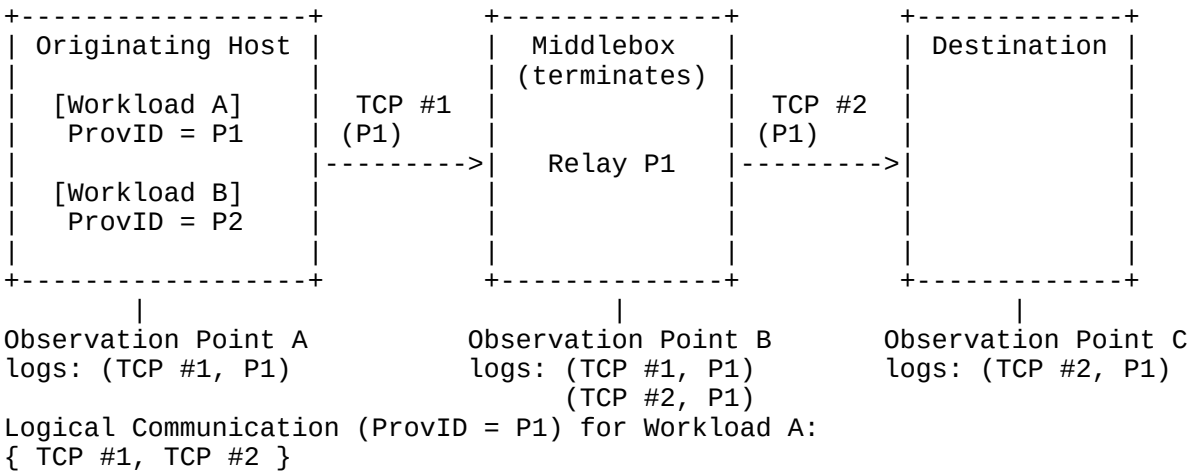

Figure 1: Provenance Correlation for a Logical Communication across Middleboxes

1.1. Requirements Language

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. Use Cases

2.1. Association of Traffic Across Rewriting

In many administrative domains, operators need to reconstruct the path of a logical communication for troubleshooting, incident investigation, or auditing. Flow logs and measurements collected at different observation points commonly reference different TCP connections that correspond to the same logical communication, and the originating endpoint cannot be reliably identified from transport

identifiers alone.

In this use case, the originating endpoint includes the ProvID in the TCP header options. Observation points record the ProvID alongside locally observed connection identifiers. When a logical communication is realized as a series of distinct TCP connections within the domain, the ProvID provides a stable correlation handle for aggregating these disparate records. This enables end-to-end reconstruction of communication paths across middleboxes, supports cross-layer observability (for example, linking network telemetry with application or process context), and maintains provenance continuity.

2.2. Process-level Origin Attribution for Remediation

An operator may detect anomalous TCP behavior at an observation point and need to remediate the issue by acting on the specific originator responsible. At that point, the original source may be obscured by address translation or connection rewriting, and even host-level attribution may be insufficient because multiple independent processes can share a host.

In this use case, the originating endpoint generates a ProvID using the pair (host IP address, process identifier) of the process that created the socket. An operator that observes abnormal traffic associated with a ProvID can map the ProvID to the initiating (host IP, process identifier) and take immediate action on that process (for example, suspend or restart the process, isolate the workload instance, or apply a narrowly scoped network policy).

3. Option Format

The Provenance Identifier (ProvID) option uses a fixed-length experimental TCP option format. The option is identified by the experimental option kind and is distinguished by a fixed option length.

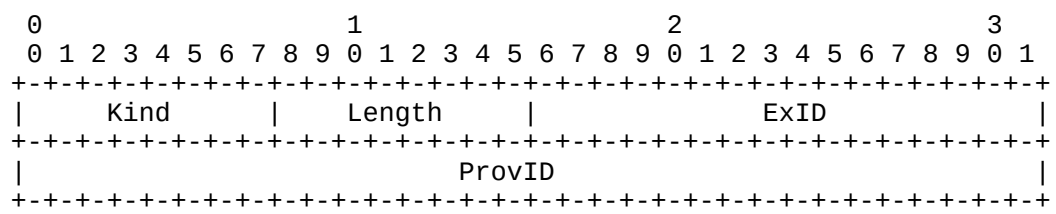

Figure 2: ProvID TCP Option Format

Kind
: The TCP option kind. The value of this field is 253.

Length
: The total length of the TCP option in bytes. For the ProvID option defined in this document, the value of this field is 12.

ExID
: The Experiment Identifier (ExID). This 2-byte field identifies the ProvID experiment when used with experimental TCP option kinds. The value of this field is 0xDEE9.

ProvID
: The Provenance Identifier. This field is 8 bytes in length and carries a provenance identifier defined by the sender.

4. Middlebox Considerations

To maintain provenance continuity within an administrative domain, middleboxes (as defined in [RFC3234]) MUST handle the ProvID option according to their function.

4.1. Non-terminating Middleboxes

A non-terminating middlebox is a device that resides on the communication path but does not terminate the end-to-end TCP connection. These middleboxes MUST forward the ProvID option unmodified in any segment where it appears. If the middlebox modifies transport-layer identifiers (e.g., performing Network Address Translation), it MUST NOT strip or alter the ProvID option.

4.2. Terminating Middleboxes

A terminating middlebox is a device that acts as the endpoint for a TCP connection (e.g., a service proxy). To maintain provenance continuity, these middleboxes MUST relay the ProvID from the incoming connection to the outgoing connection. Specifically, whenever a ProvID is observed in an incoming TCP segment, the middlebox MUST include the identical ProvID value in the corresponding segment of the outgoing connection.

4.3. Domain Boundary Handling

To prevent the leakage of internal network metadata, middleboxes at the boundary of the administrative domain MUST strip the ProvID option from any TCP segments exiting the domain. Similarly, any ProvID options present in traffic entering the domain from the open Internet MUST be stripped.

5. IANA Considerations

This document's IANA considerations are to be determined and will be provided in a subsequent revision of this draft. [TODO]

6. Security Considerations

This document's security considerations are to be determined and will be provided in a subsequent revision of this draft. [TODO]

7. References

7.1. 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>.

[RFC6994] Touch, J., "Shared Use of Experimental TCP Options", RFC 6994, DOI 10.17487/RFC6994, August 2013, <https://www.rfc-editor.org/info/rfc6994>.

[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>.

[RFC9293] Eddy, W., Ed., "Transmission Control Protocol (TCP)", STD 7, RFC 9293, DOI 10.17487/RFC9293, August 2022, <https://www.rfc-editor.org/info/rfc9293>.

[RFC3234] Carpenter, B. and S. Brim, "Middleboxes: Taxonomy and Issues", RFC 3234, DOI 10.17487/RFC3234, February 2002, <https://www.rfc-editor.org/info/rfc3234>.

7.2. Informative References

Appendix A. Appendix 1

TODO

Authors' Addresses

Bowen Liang
Tsinghua University
Email: liangbw25@mails.tsinghua.edu.cn

Yang Xiang
Yunshan Networks
Email: xiangyang@yunshan.net

Xingang Shi
Tsinghua University
Email: shixg@cernet.edu.cn

Xia Yin
Tsinghua University
Email: yinxia@tsinghua.edu.cn