Standard Communication with Network Elements
Internet-Draft
Intended status: Informational
Expires: 9 August 2025

M. Thomson
Mozilla

5 February 2025

A comparative analysis of SCONE relative to TRAIN
draft-thomson-scone-merge-criticisms-00

Abstract

A merge of the rate availability signaling schemes in the SCONE and TRAIN proposals is analysed and found to be infeasible.

About This Document

This note is to be removed before publishing as an RFC.

The latest revision of this draft can be found at https://martinthomson.github.io/scone-merge-criticisms/draft-thomson-scone-merge-criticisms.html. Status information for this document may be found at https://datatracker.ietf.org/doc/draft-thomson-scone-merge-criticisms/.

Discussion of this document takes place on the Standard Communication with Network Elements mailing list (mailto:scone@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/scone. Subscribe at https://www.ietf.org/mailman/listinfo/scone/.

Source for this draft and an issue tracker can be found at https://github.com/martinthomson/scone-merge-criticisms.

Status of This Memo

Table of Contents

1. Introduction 2
2. Communication Model 3
2.1. Analysis 3
3. Signal 3
3.1. Analysis 3
4. Network Element Processing 4
4.1. Analysis 4
5. Miscellaneous SCONE Problems 4
5.1. Packet Protection 5
5.2. Forwarding Bit 5
5.3. Source Connection ID on Signals 6
6. Security Considerations 6
7. IANA Considerations 6
8. Informative References 6
Acknowledgments 6
Author's Address 6

1. Introduction

There are a few items where we might need to discuss the advantages of SCONE [SCONE], but overall there are not a lot of advantages to the proposed design over the one that TRAIN [TRAIN] proposes.

The idea that these designs might be merged does not make a lot of sense. The two designs are largely incompatible, as each relies on a fundamentally different mechanism.

The one major question about the nature of the signal is separable from a decision about how to design the signaling mechanism.

2. Communication Model

TRAIN is symmetric. SCONE is initiated by a client only.

TRAIN is negotiated as part of a connection. SCONE is unilaterally implemented by clients.

TRAIN endpoints coalesce a carriage packet with other packets, network elements modify the payload of that carriage.

SCONE requires that clients announce their willingness to receive signals to network elements. Thereafter network elements can send a signal any time to that client.

2.1. Analysis

Like ECN, TRAIN signals follow the flow in the direction in which it might be affected. SCONE signals flow in the reverse direction, which means that if flows take a different path in each direction, the wrong set of network elements are involved.

SCONE packets have far weaker authentication. Once a packet is captured, any entity can generate a packet that will be accepted. In comparison, a TRAIN packet is carried along with real packets and so is only good for approximately as long as it takes to get to its destination. Spoofing requires that the packet be raced.

The requirement for negotiation in TRAIN makes it a tiny bit more complex. However, it also means that both client and server agree to its use before it is used. Negotiation is necessary to enable coalescing in TRAIN. Whether this is an advantage to TRAIN or SCONE seems subjective.

Arguably that a network element can send an updated signal at any time is an advantage for SCONE.

3. Signal

TRAIN presently proposes a signal that carries a choice from an (as-yet-unspecified) selection of 64 options. That's not a lot.

SCONE has a 32-bit rate (in kbps) and a window (in ms), which is massively more flexible.

3.1. Analysis

This is not a serious difference. TRAIN could change to accommodate a richer signal, but that comes with costs, see Section 4.

4. Network Element Processing

TRAIN requires processing similar to ECN: network elements recognize opportunities to send signals when they receive packets for forwarding. Sending a signal involves the flipping of a few bits. This is designed to require no state and be as simple as possible to process, such that it could be done easily at line rate. This is a strong reason to limit the expressiveness of the signal: a more complex signal would be harder to apply.

SCONE involves establishing state at a network element, with the network element sending a signal any time it chooses. Generating the signal requires the use of an AEAD. In practice, this is likely a few packets for new flows, then a few any time the flow characteristics at the element change.

4.1. Analysis

SCONE requires that network elements generate new packets, including spoofing the source IP address. TRAIN only depends on packet modification.

SCONE requires that network elements remember an address tuple and client connection ID so that they can provide updates. TRAIN can be processed without any state.

> | Elements might benefit from recognizing a flow as QUIC and
> | remembering it to manage this issue
> | (https://github.com/martinthomson/train-protocol/issues/32).
> | However, that issue might be resolved by a modest protocol
> | change instead.

SCONE exposes network elements to weird attacks if they don't maintain additional state. They can be sent SCONE packets without an associated QUIC flow, which might cause the network element to spend effort on sending packets to a spoofed source address. Given spoofed SCONE packets, those generated packets go to a destination of the attacker's choice. The time frame over which those packets are likely spread means that this is unlikely to be a significant amplification in terms of packet rate. That might make DoS using this vulnerability less appealing, but it is worth noting as it has no upper bound in quantity.

5. Miscellaneous SCONE Problems

There are a few things in SCONE that probably could be improved. None of these really have a bearing on the overall solution.

5.1. Packet Protection

SCONE defines packet protection for the packet that a client sends. Network elements do not need to decrypt or authenticate packets the information they need from that packet is not protected but that's not obvious from the specification.

Protection of signals from network elements provides no meaningful security value. The value is in resilience to accidentally interpreting garbage packets as a signal and the AEAD is definitely better than the UDP checksum at detecting transmission errors. However, those same benefits might be realized more cheaply by other means.

5.2. Forwarding Bit

SCONE packets include a forward bit, which if set requests that network elements drop the packet. This requires exceptional processing by a network element, other than simply recognizing the packet. That requirement might take all SCONE packets of the fast path in elements.

It seems like the point of this is to enable the targeting of a first-hop network element. However, this assumes that the application of rate limits is done by something on that first network segment. That's true in a lot of cases, but this is a bad assumption. The same might be more easily achieved by altering the IP TTL instead, with greater targeting precision.

Alternatively, this feature might be used to probe whether a path supports SCONE: if a forward=0 packet makes it to the other end, it might feed that information back. Two things though:

* This feature on its own probably wouldn't be enough to justify a server implementation.

* Network elements aren't required to drop these packets.

TRAIN gets this capability for free. An endpoint that receives an unmodified TRAIN packet might infer that the path doesn't support TRAIN.

5.3. Source Connection ID on Signals

SCONE insists on a random source connection ID from network elements. To start with, this might cause things to drop the signals if they expect a particular source connection ID. Also, as there is no ongoing communication, this could just swap source and destination connection ID from the original. That would strengthen protection against spoofing (many clients do not use connection IDs for QUIC packets they receive) at a small cost in maintained state. However, any spoofing protection thus gained would not be assured, as there is no minimum entropy for connection IDs in either direction.

6. Security Considerations

Requiring this section is perhaps no longer sensible, when due consideration is given to the topic of security throughout, as is the case with this document.

7. IANA Considerations

This document has no IANA actions.

8. Informative References

[SCONE] Joras, M. and L. M. Ihlar, "A new QUIC version for network property communication", Work in Progress, Internet-Draft, draft-joras-scone-quic-protocol-00, 21 October 2024, <https://datatracker.ietf.org/doc/html/draft-joras-scone-quic-protocol-00>.

[TRAIN] Thomson, M., Huitema, C., and K. Oku, "Transparent Rate Adaptation Indications for Networks (TRAIN) Protocol", Work in Progress, Internet-Draft, draft-thomson-scone-train-protocol-00, 14 October 2024, <https://datatracker.ietf.org/doc/html/draft-thomson-scone-train-protocol-00>.

Acknowledgments

Christian Huitema provided useful feedback, but does not necessarily endorse its message.

Author’s Address

Martin Thomson
Mozilla
Email: mt@lowentropy.net