MMUSIC
Internet-Draft
Intended status: Informational
Expires: May 4, 2020

A. Drake
J. Uberti
Q. Wang
Google

November 01, 2019

Encrypting ICE candidates to improve privacy and connectivity
draft-wang-mmusic-encrypted-ice-candidates-00

Abstract

WebRTC applications collect ICE candidates as part of the process of creating peer-to-peer connections. To maximize the probability of a direct peer-to-peer connection, client private IP addresses can be included in this candidate collection, but this has privacy implications. This document describes a way to share local IP addresses with local peers without compromising client privacy. During the ICE process, local IP addresses are encrypted and authenticated using a pre-shared key and cipher suite before being put into ICE candidates as hostnames with an ".encrypted" pseudo-top-level domain. Other peers who also have the PSK are able to decrypt these addresses and use them normally in ICE processing.

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 http://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 4, 2020.

Table of Contents

1. Introduction

The technique detailed in [MdnsCandidate] provides a method to share local IP addresses with other clients without exposing client private IP to applications. Given the fact that the application may control the signaling servers, STUN/TURN servers, and even the remote peer implementation, the locality of the out-of-band mDNS signaling can be considered the sole source of trust between peers to share local IPs. However, mDNS messages are by default scoped to local links [RFC6762], and may not be enabled to traverse subnets in certain networking environments. These environments may experience frequent failures in mDNS name resolution and significant connectivity challenges as a result. On the other hand, endpoints in these environments are typically managed, in such a way that information can be securely pushed and shared, including a pre-shared key and its associated cipher suite.

This document proposes a complementary solution for managed networks to share local IP addresses over the signaling channel without compromising client privacy. Specifically, addresses are encrypted with pre-shared key (PSK) cipher suites, and encoded as hostnames with the ".encrypted" pseudo-top-level domain (pseudo-TLD).

WebRTC and WebRTC-compatible endpoints [Overview] that receive ICE candidates with encrypted addresses will authenticate these hostnames in ciphertext, decrypt them to IP addresses, and perform ICE processing as usual. In the case where the endpoint is a web application, the WebRTC implementation will manage this process internally and will not disclose the IP addresses in plaintext to the application.

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. Description

This section uses the concept of ICE agent as defined in [RFC8445].

3.1. Pre-Shared Key Cipher Suite

ICE agents that implement this proposal pre-share keys for cipher suites based on symmetric-key algorithms. The mechanism of sharing such information is outside the scope of this document, but viable mechanisms exist in browsers today.

The implementation MUST support the Advanced Encryption Standard (AES) [AES] algorithm and its operation in the CTR, CBC or GCM mode with message authentication, and SHOULD use the GCM mode whenever it is supported. The implementation MUST pre-determine a single mode to use as part of the mechanism to share the information about the cipher suite. When using the CTR or CBC mode, HMAC with SHA-2 MUST be supported.

Since the plaintext to encrypt consists of only a single IPv4 or IPv6 address that fits in a single 128-bit block, the initialization parameter for each mode can be a cryptographically random number. In particular, this parameter is given by a 16-byte initial counter block value for CTR, or a 16-byte or 12-byte initialization vector for CBC or GCM, respectively.

Note the ICE password associated with an ICE agent has at least 128-bit randomness as defined by [RFC8445]. To reduce the overhead in the candidate encoding that will be detailed in the next section, the initialization parameter MUST be chosen as the first 16 bytes or 12 bytes in the network order for the mode in use.

3.2. ICE Candidate Gathering

This section outlines how a PSK cipher suite should be used by ICE agents to conceal local IP addresses.

3.2.1. Procedure

For each host candidate gathered by an ICE agent as part of the gathering process described in [RFC8445], Section 5.1.1, the candidate is handled as described below.

1. Check whether the IP address satisfies the ICE agent's policy regarding whether an address is safe to expose. If so, expose the candidate and abort this process.

2. Generate the encrypted address.

    1. Let _address_ be the IP address of the candidate, and embed it as an IPv6 address if it is an IPv4 address, using the "Well-Known Prefix" as described in [RFC6052].

    2. Let _ciphersuite_ be the pre-determined cipher suite and its initialization parameter, and _key_ the PSK.

    3. Let _EncryptAndAuthenticate(plaintext, ciphersuite, key)_ be an operation that uses the given cipher suite to encrypt a given plaintext with authentication, and returns concatenated ciphertext and message authentication code (MAC).

    4. Compute _encrypted_address_ as the output of _EncryptAndAuthenticate(address, ciphersuite, key)_.

3. Generate a pseudo-FQDN as follows.

    1. Encode _encrypted_address_ to a hex string, and split the hex string to substrings after every 32 characters.

    2. Form a string by joining the substrings above sequentially with the delimiter ".". Denote the formed string by _encoded_encrytped_address_.

    3. Generate the pseudo-FQDN "_encoded_encrypted_address.encrypted_" with the pseudo-TLD "_.encrypted_".

4. Replace the IP address of the ICE candidate with the pseudo-FQDN from step 3, and provide the candidate to the application.

3.2.2. Example

The candidate attribute in an SDP message to exchange the encrypted candidate can be given by

a=candidate:1 1 udp 2122262783 8c9bd03bb7a5a76a5803eebc688f0388.fa991acbdf116f6b72fd3a781174cd58.encrypted 56622 typ host

following the above procedure. This example assumes the use of the GCM mode, in which case the 256-bit _encrypted_address_ consists of 128-bit ciphertext and 128-bit MAC, and can be encoded to 64 hex characters as two labels.

3.3. ICE Candidate Processing

This section outlines how received ICE candidates with mDNS names are processed by ICE agents, and is relevant to all endpoints.

For any remote ICE candidate received by the ICE agent, the following procedure is used.

1. If the connection-address field value of the ICE candidate does not end with ".encrypted", then process the candidate as defined in [RFC8445] or [MdnsCandidate].

2. If the ICE agent has no PSK cipher suite for encrypted candidates, proceed to step 5.

3. Decrypt the address as follows.

    1. Let _AuthenticateAndDecrypt(ciphertext_and_mac, ciphersuite, key)_ be an operation using the given cipher suite to authenticate and decrypt a given ciphertext with MAC, and returns the decrypted value, or an fail-to-decrypt (FTD) error.

    2. Let _encoded_encrypted_address_ be the value of the connection-address field after removing the trailing "_.encrypted_", and let _encrypted_address_ be the string after removing all "." in _encoded_encrypted_address_.

    3. Let _decrypted_address_ be given by _AuthenticateAndDecrypt(encrypted_address)_. If _decrypted_address_ does not represent a valid IPv6 address or an embedded IPv4 address, or an FTD error is raised, proceed to step 5.

4. Convert _decrypted_address_ to an IPv4 address if it is embedded.

4. Replace the connection-address field of the ICE candidate with _decrypted_address_, skip the rest steps and continue processing of the candidate as described in [RFC8445].

5. Discard the candidate, or proceed to step 6 if the ICE agent implements [MdnsCandidate].

6. Let _encoded_encrypted_address_ be the same value as defined in step 3. Construct an mDNS name given by "_encoded_encrypted_address.local_", and proceed to step 2 in Section 3.2.1 in [MdnsCandidate].

ICE agents can implement this procedure in any way as long as it produces equivalent results.

4. Security Considerations

4.1. mDNS Message Flooding via Fallback Resolution

Encrypted candidates can be spoofed and signaled to an ICE agent to trigger the fallback mDNS resolution as described in step 6 in Section 3.3. This can potentially generate excessive traffic in the subnet. Note however that implementations of [MdnsCandidate] are required to have a proper rate limiting scheme of mDNS messages.

5. IANA Considerations

This document requires no actions from IANA.

6. References

6.1. 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>.

[RFC6052] Bao, C., Huitema, C., Bagnulo, M., Boucadair, M., and X. Li, "IPv6 Addressing of IPv4/IPv6 Translators", RFC 6052, DOI 10.17487/RFC6052, October 2010, <https://www.rfc-editor.org/info/rfc6052>.

[RFC6762] Cheshire, S. and M. Krochmal, "Multicast DNS", RFC 6762, DOI 10.17487/RFC6762, February 2013, <https://www.rfc-editor.org/info/rfc6762>.

[RFC8445] Keranen, A., Holmberg, C., and J. Rosenberg, "Interactive Connectivity Establishment (ICE): A Protocol for Network Address Translator (NAT) Traversal", RFC 8445, DOI 10.17487/RFC8445, July 2018, <https://www.rfc-editor.org/info/rfc8445>.

6.2. Informative References

[AES] National Institute of Standards and Technology, "Specification for the Advanced Encryption Standard (AES)", FIPS 197, November 2001.

[MdnsCandidate] Wang, Q., "Using Multicast DNS to protect privacy when exposing ICE candidates", October 2019, <https://tools.ietf.org/html/draft-ietf-rtcweb-mdns-ice-candidates>.

[Overview] Alvestrand, H., "Overview: Real Time Protocols for Browser-based Applications", November 2017, <https://tools.ietf.org/html/draft-ietf-rtcweb-overview>.

Authors' Addresses

Alex Drake
Google

Email: alexdrake@google.com

Justin Uberti
Google

Email: juberti@google.com

Qingsi Wang
Google

Email: qingsi@google.com