RTCWEB
Internet-Draft
Intended status: Informational
Expires: December 31, 2018

Y. Fablet
Apple Inc.

June 29, 2018

Using Multicast DNS to protect privacy when exposing ICE candidates
draft-mdns-ice-candidates-00

Abstract

WebRTC applications rely on ICE candidates to enable peer-to-peer connections between clients in as many network configurations as possible. To maximize the probability to create a direct peer-to-peer connection, client private IP addresses are often exposed without user consent. This is currently used as a way to track users. This document describes a way to share IP addresses with other clients while preserving client privacy. This is achieved by obfuscating IP addresses using dynamically generated names resolvable through Multicast DNS [RFC6763].

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 December 31, 2018.

Table of Contents

1. Introduction

As detailed in [IPHandling], exposing client private IP addresses allows maximizing the probability to successfully create a connection between two clients. This information is also used by many web sites as a way to fingerprint and identify users without their consent.

The first approach exposes client private IP addresses by default, as can be seen from websites such as [IPLeak]. The second approach implemented in the WebKit engine enforces the following policy:

1. By default, use mode 3 as defined in [IPHandling]: any host ICE candidate is filtered out.

2. Use mode 2 as defined in [IPHandling] if there is an explicit user action to trust the web site: host ICE candidates are exposed to the web site based on the use of navigator.mediaDevices.getUserMedia, which typically prompts the user to grant or deny access to cameras/microphones.

The second approach supports most common audio/video conference applications but leads to failing or suboptimal connections for applications relying solely on data channel. This is particularly an issue on unmanaged networks, typically home or small offices where NAT loopback might not be supported.

To overcome the shortcomings of the above two approaches, this document proposes to register dynamically generated names using

Multicast DNS when gathering ICE candidates. These dynamically generated names are used to replace private IP addresses in host ICE candidates. Only clients that can resolve these dynamically generated names using Multicast DNS will get access to the actual client IP address.

2. Privacy Concerns

The gathering of ICE candidates without user consent is a well-known fingerprinting technique to track users. This is particularly a concern when users are connected through a NAT which is a usual configuration. In such a case, knowing both the private IP address and the public IP address will usually identify uniquely the user device. Additionally, Internet web sites can more easily attack intranet web sites when knowing the intranet IP address range.

A successful WebRTC connection between two peers is also a potential thread to user privacy. When a WebRTC connection latency is close to zero, the probability is high that the two peers are running on the same device. Browsers often isolate contexts one from the other. Private browsing mode contexts usually do not share any information with regular browsing contexts. The WebKit engine isolates third-party iframes in various ways (cookies, ITP) to prevent user tracking. Enabling a web application to determine that two contexts run in the same device would defeat some of the protections provided by modern browsers.

3. Principle

This section uses the concept of ICE agent as define in [RFC5245]. In the remainder of the document, it is assumed that each browser execution context has its own ICE agent.

3.1. ICE Candidate Gathering

For any host ICE candidate gathered by a browsing context as part of [RFC5245] section 4.1.1, obfuscation of the candidate is done as follows:

1. Check whether the context ICE agent registered a name resolving to the ICE host candidate IP address.

2. If the ICE agent registered the name, replace the IP address of the ICE host candidate with the name with ".local" appended to it. Expose the candidate and abort these steps.

3. Generate a random unique name, typically a version 4 UUID as defined in [RFC4122].

4. Register the unique name using Multicast DNS.

5. If registering of the unique name fails, abort these steps. The candidate is not exposed.

6. Store the name and its related IP address in the ICE agent for future reuse.

7. Replace the IP address of the ICE host candidate with the name with ".local" appended to it. Expose the candidate.

3.2. ICE Candidate Processing

For any remote host 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 finish by ".local", process the candidate as defined in [RFC5245].

2. Otherwise, remove the ".local" suffix to the value and resolve it using Multicast DNS.

3. If it resolves to an IP address, replace the value of the ICE host candidate by the resolved IP address and continue processing of the candidate.

4. Otherwise, ignore the candidate.

Multicast DNS resolution might end up retrieving both an IPv4 and IPv6 address. In that case, the IPv6 address may be used preferably to the IPv4 address.

4. Privacy Guidelines

4.1. APIs leaking IP addresses

When there is no user consent, the following filtering should be done to prevent private IP address leakage:

1. host ICE candidates with an IP address are not exposed as ICE candidate events.

2. Server reflexive ICE candidate raddr field is set to 0.0.0.0 and rport to 0.

3. SDP does not expose any a=candidate line corresponding to a host ICE candidate which contains an IP address.

4. RTCIceCandidateStats dictionaries exposed to web pages do not contain any 'ip' member if related to a host ICE candidate.

4.2. Generated names reuse

Dynamically generated names can be used to track users if used too often. Conversely, registering too many names will also generate useless processing. The proposed rule is to create and register a new generated name for a given IP address on a per execution context.

4.3. Specific execution contexts

Privacy might also be breached if two execution contexts can identify whether they are run in the same device based on a successful peer-to-peer connection. The proposed rule is to not register any name using Multicast DNS for any ICE agent belonging to:

1. A third-party browser execution context, i.e. a context that is not same origin as the top level execution context.

2. A private browsing execution context.

5. Specification Requirements

The proposal relies on identifying and resolving any Multicast DNS based ICE candidates as part of adding/processing a remote candidate. [ICESDP] section 4.1 could be updated to explicitly allow Multicast DNS names in the connection-address field.

The proposal relies on adding the ability to register Multicast DNS names at ICE gathering time. This could be described in [ICESDP] and/or [WebRTCSpec].

The proposal allows updating [IPHandling] so that mode 2 is not the mode used by default when user consent is not required. Instead, the default mode could be defined as mode 3 with Multicast DNS based ICE candidates.

6. Informative References

[ICESDP] Keranen, A., "Session Description Protocol (SDP) Offer/ Answer procedures for Interactive Connectivity Establishment (ICE)", April 2018, <https://tools.ietf.org/html/ draft-ietf-mmusic-ice-sip-sdp>.

[IPHandling]
Shieh, G., "WebRTC IP Address Handling Requirements", April 2018, <https://tools.ietf.org/html/draft-ietf-rtcweb-ip-handling>.

[IPLeak] "IP/DNS Detect", n.d., <https://ipleak.net>.

[RFC4122] Leach, P., Mealling, M., and R. Salz, "A Universally Unique IDentifier (UUID) URN Namespace", RFC 4122, DOI 10.17487/RFC4122, July 2005, <https://www.rfc-editor.org/info/rfc4122>.

[RFC5245] Rosenberg, J., "Interactive Connectivity Establishment (ICE): A Protocol for Network Address Translator (NAT) Traversal for Offer/Answer Protocols", RFC 5245, DOI 10.17487/RFC5245, April 2010, <https://www.rfc-editor.org/info/rfc5245>.

[RFC6763] Cheshire, S. and M. Krochmal, "DNS-Based Service Discovery", RFC 6763, DOI 10.17487/RFC6763, February 2013, <https://www.rfc-editor.org/info/rfc6763>.

[WebRTCSpec]
Bruaroey, J., "The WebRTC specification", n.d., <https://w3c.github.io/webrtc-pc/>.

Author’s Address

Youenn Fablet
Apple Inc.

Email: youenn@apple.com