mDNS ICE Candidates

J. de Borst, Y. Fablet, J. Uberti, Q. Wang

IETF 103

Purpose

- WebRTC by default exposes host candidates to web pages
  - To enable the most efficient connection path
- This information is used by web pages to fingerprint users
  - Gathering of private IPv4 addresses
- Chrome, Firefox and Edge do expose default route host candidates by default
- Safari does not expose any host candidates by default
  - This hurts connection success/connection efficiency

Specifications Scope

- IP-Handling v1
- mDNS ICE candidates
  - Define the technique to use mDNS for ICE candidates
- IP-Handling v2
  - Integrate new mode(s) based on mDNS ICE candidates proposal

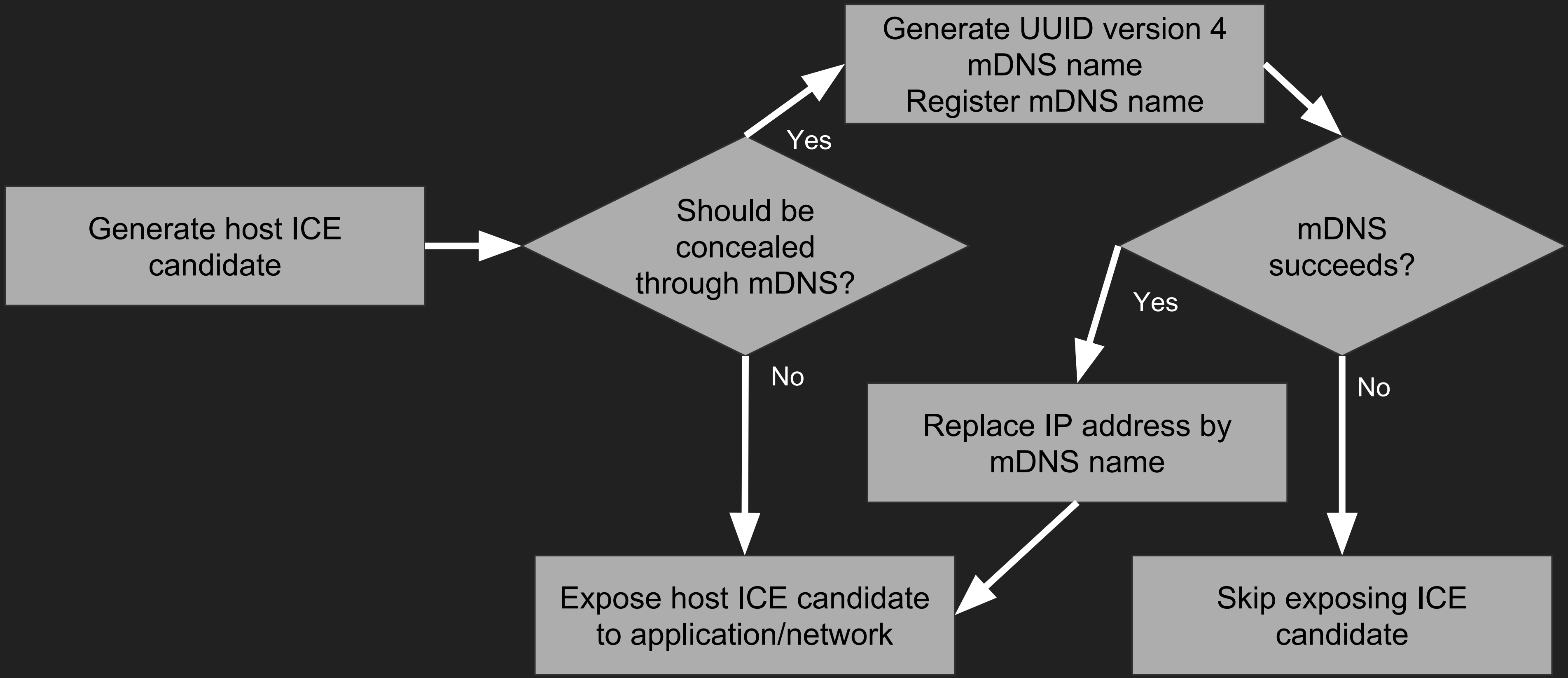
IP-Handling v1

- Improve the description of private IP addresses issue
- Mention the possibility for future modes
- Leave other work for future documents

mDNS ICE Candidates Draft

- Active development on GitHub
  - https://github.com/youennf/mdns-ice-candidates

Candidate Generation



Generate host ICE candidate
Should be concealed through mDNS?
Yes
Generate UUID version 4 mDNS name
Register mDNS name
mDNS succeeds?
Yes
Replace IP address by mDNS name
No
Skip exposing ICE candidate
No
Expose host ICE candidate to application/network

When to Use mDNS for Host Candidates?

- Concealment is not needed for public IP addresses
- IPv4/IPv6 STUN servers to the rescue
  - Send mDNS candidates as soon as possible
  - Also send server-reflexive candidates when computed
    - Even if the mDNS candidate conceals the public IP address exposed by the server-reflexive candidate
- Possibility to store whether an address is public or private from past interactions

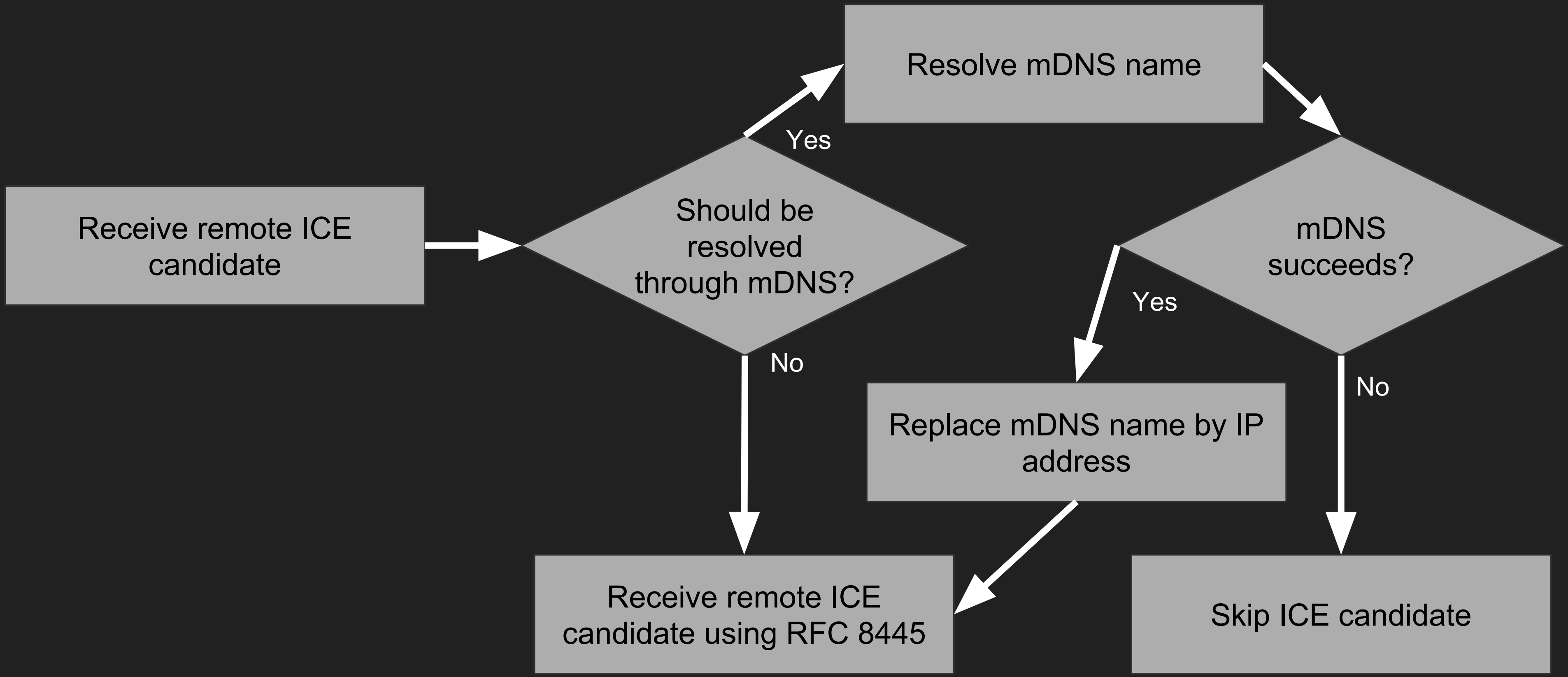
mDNS Name Reuse

- mDNS names should be limited in time and scope
  - Otherwise these names might become even better fingerprints than the IP addresses they conceal
- Solution
  - Scope by origin of the web page
  - Limit lifetime to the life of the web page

Candidate Generation Additional Points

- Implementation target
  - Browsers
  - Endpoints wary of exposing information about their network
- Consistent concealment
  - mDNS names should be used consistently in ICE Candidates, SDP, WebRTC stats
  - Server-reflexive candidates should be filtered
    - (rdar, rport) = (0.0.0.0, 0)

Candidate Resolution



Receive remote ICE candidate

Should be resolved through mDNS?

Yes

Resolve mDNS name

mDNS succeeds?

Yes

Replace mDNS name by IP address

No

Skip ICE candidate

No

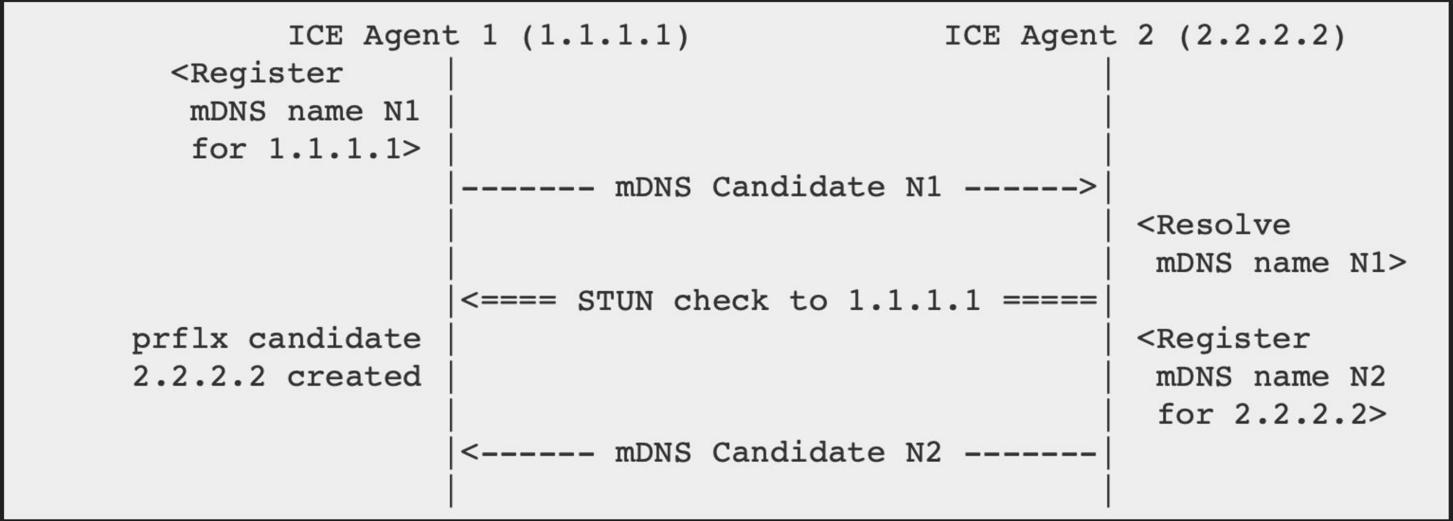
Receive remote ICE candidate using RFC 8445

Candidate Resolution Additional Points

- Implementation target
  - All endpoints implementing ICE
- When to use mDNS resolution
  - Name ending with ‘.local’
  - May be restricted to only version 4 UUID names
- Multiple IPs for a single mDNS name?
  - Proposed behavior
    - Select a single address, first IPv6 if available
  - Should not happen in practice
    - Registration mandates one name per IP address

WebRTC Stats IP Leakage

- Exposure of peer-reflexive IP addresses through RTCIceCandidateStats

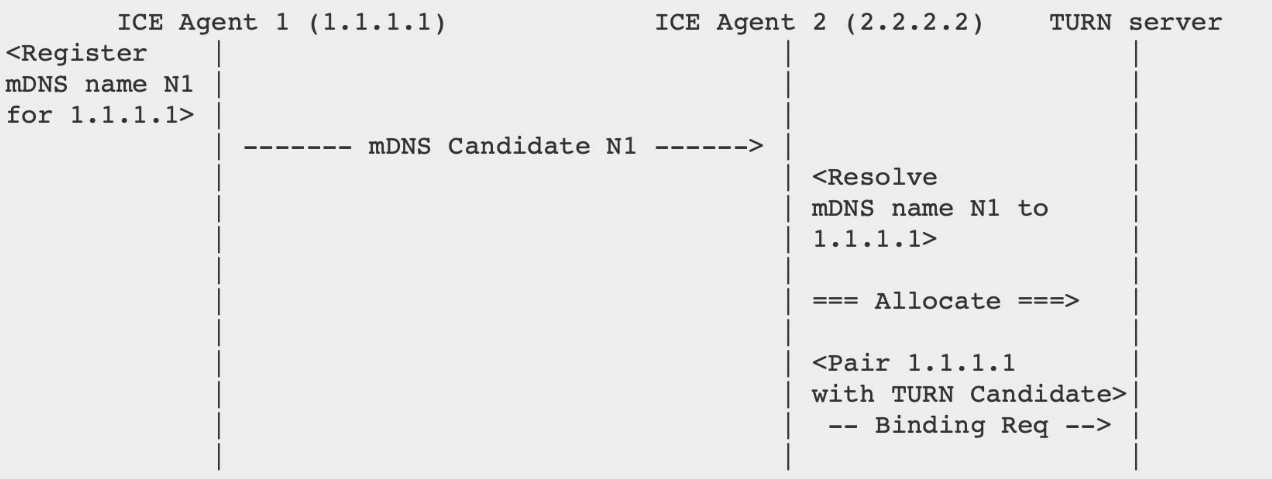


```
            ICE Agent 1 (1.1.1.1)             ICE Agent 2 (2.2.2.2)
      <Register     |                                 |
       mDNS name N1 |                                 |
       for 1.1.1.1> |                                 |
                    |------- mDNS Candidate N1 ------>|
                    |                                 | <Resolve
                    |                                 |  mDNS name N1>
                    |<==== STUN check to 1.1.1.1 =====|
    prflx candidate |                                 | <Register
    2.2.2.2 created |                                 |  mDNS name N2
                    |                                 |  for 2.2.2.2>
                    |<------ mDNS Candidate N2 -------|
                    |                                 |
```

- No exposure of peer-reflexive candidate addresses in WebRTC stats
  - Unless already known by web application

TURN Server IP Leakage

- Destination IP addresses are sent to relay servers when generating relay candidate pairs
    - This would defeat mDNS obfuscation



```
       ICE Agent 1 (1.1.1.1)             ICE Agent 2 (2.2.2.2)    TURN server
<Register    |                                   |                     |
mDNS name N1 |                                   |                     |
for 1.1.1.1> |                                   |                     |
             | ------- mDNS Candidate N1 ------> |                     |
             |                                   | <Resolve            |
             |                                   | mDNS name N1 to     |
             |                                   | 1.1.1.1>            |
             |                                   |                     |
             |                                   | === Allocate ===>   |
             |                                   |                     |
             |                                   | <Pair 1.1.1.1       |
             |                                   | with TURN Candidate>|
             |                                   |  -- Binding Req --> |
             |                                   |                     |
```

- Solution
    - Do not use remote mDNS candidates to pair relay candidates
    - No impact on connectivity

Network Interface Enumeration

- Number of mDNS candidates as a fingerprinting method
  - Not an issue if limited to default route candidates
- Proposal
  - Reconsider this issue if/when exposing non-default route mDNS candidates
  - Limit the number and/or variability of candidates

mDNS Message Flooding

- Flooding with mDNS traffic by web pages
    - Both registration and resolution
- Proposal
    - Limit resolution requests as per RFC 6762
    - Make browsers throttle registrations

mDNS Name Denial

- Malicious endpoints in the local network can break mDNS registration/resolution
    - May limit direct connectivity
- Proposal
    - Outside of the scope of this document

Reduced Connectivity

- mDNS resolution might fail
  - Networks not supporting mDNS
  - Endpoints too far away on the same large network
- Proposal
  - Gather experimental data to fully assess the severity of the issue
  - Investigate solutions in addition to NAT hairpinning and TURN
    - Bypass mDNS concealment for IPv6 RFC 4941/7217 addresses
    - DNS-SD mDNS relays

Connection Setup Latency

- Registration & resolution might affect connection setup latency
- Proposal
  - Gather experimental data
    - Local network should be fast in most cases
  - Implementations may decide to not wait for registration success to send the corresponding ICE candidate
  - Possibility to pre-register mDNS names

Backward Compatibility

- Legacy endpoints might not resolve mDNS ICE Candidates
  - Or resolve them through DNS
- But
  - Legacy endpoints will probably expose their host candidates which should allow direct connection

Implementation Support

- LibWebRTC
  - Full support of registration and resolution
- WebKit/Safari Technology Preview
  - Full support of registration and resolution, the latter based on libwebrtc
  - Experimental feature turned off by default
- Chrome
  - Full support of registration and resolution
  - Available in Canary Windows & Linux
    - Enable using **`chrome://flags`**

Empirical Data Gathering plan

- Measure the drop in connection success
    - Gather success rate for mDNS-enabled to mDNS-enabled connections
    - Gather success rate for mDNS-disabled to mDNS-disabled connections
    - Compare the two success rates
    - Need to make sure that there is an even distribution of mDNS-enabled/mDNS-disabled endpoints
- Measure connection latency increase

IP Handling v2

- Main target
  - New mode(s) between mode 2 and mode 3
  - https://tools.ietf.org/html/draft-uberti-ip-handling-ex-mdns-00
- Potential future target
  - New mode(s) between mode 1 and mode 2
  - Expose non-default route candidates
  - Need to investigate potential fingerprinting issues