rtcweb-ietf-ip-handling-01

Justin Uberti

Refresher: Goals

- Prevent drive-by address harvesting, especially ISP IP when using VPN
- Avoid degrading user experience or quality by default
- Provide options to prevent exposure of local IP addresses and force use of proxy in specific cases

Refresher: 4 Modes

1. **Everything**
   (default, with consent)
2. **Restricted gathering, single host candidate**
   (default, no consent)
3. **Restricted gathering, no host candidates**
   (via prefs or extension)
4. **Force proxy**
   (via prefs or extension)

Changes from individual draft

- Detailed reviews by Adam Roach and Wendy Seltzer (thanks!)
- Revamped controversial section on coupling IP gathering permission with cam/mic permission
- Discussion of proxies now considers non-TCP and RETURN proxies
- Added necessary references
- Various editorial changes

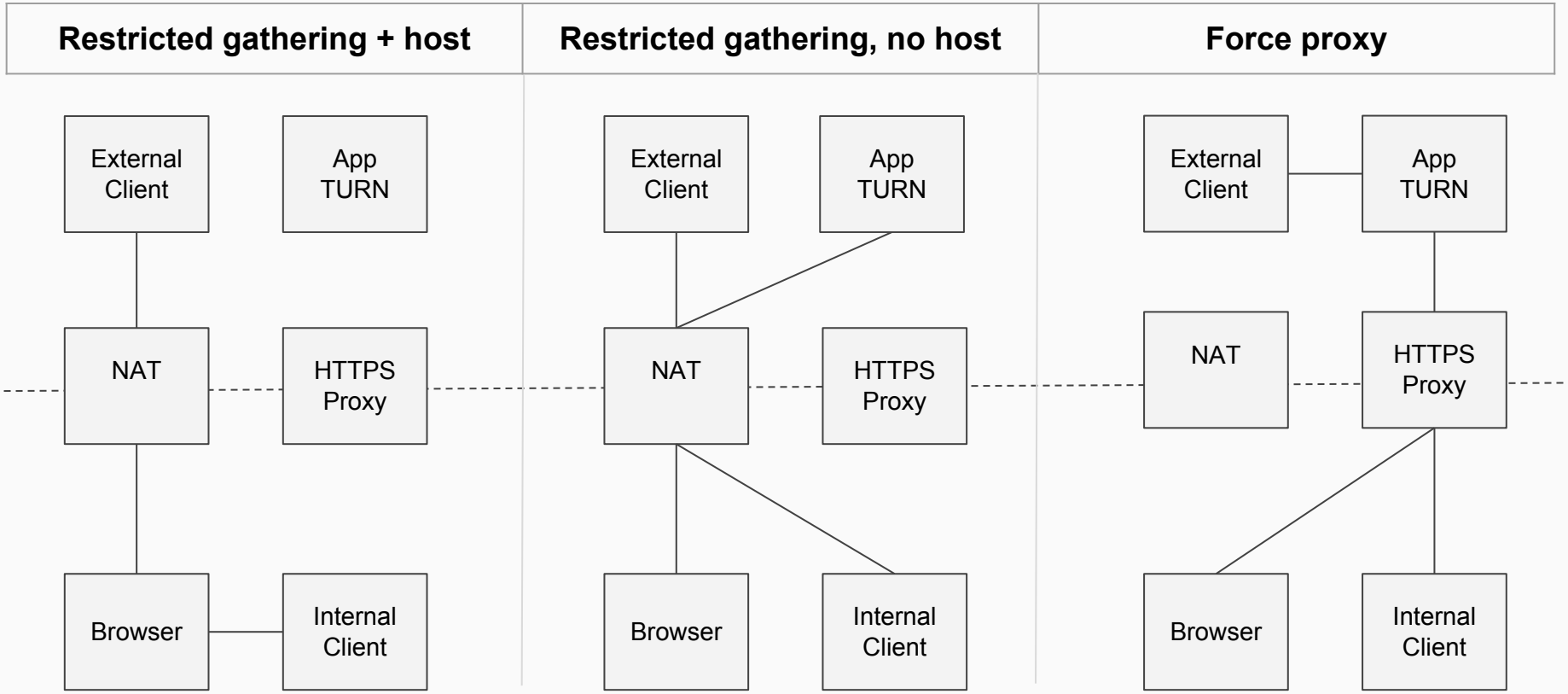
IP Gathering Permission

Old	New
WebRTC incorporates an explicit permission grant for access to local audio and video, **which are typically much more sensitive than the aforementioned IP address information**. If the user has consented to media access, this should also allow WebRTC to gather all possible candidates and determine the absolute best route for media traffic.	When used with audio and video devices, WebRTC requires explicit user permission to access those devices. We propose that this permission grant be expanded to include consent to allow WebRTC to access all IP addresses associated with the user agent, for the purpose of finding the absolute best route for media traffic. **Combining these permission grants, rather than having the user grant permission individually, is a considered balance**; this balance takes into account that the user has placed enough trust into the application to allow it to access their devices, that when doing so the user typically wants to engage in a conversational session, which benefits most from an optimal network path, and lastly, the fact that the underlying issue is complex, and difficult to explain meaningfully to the user.

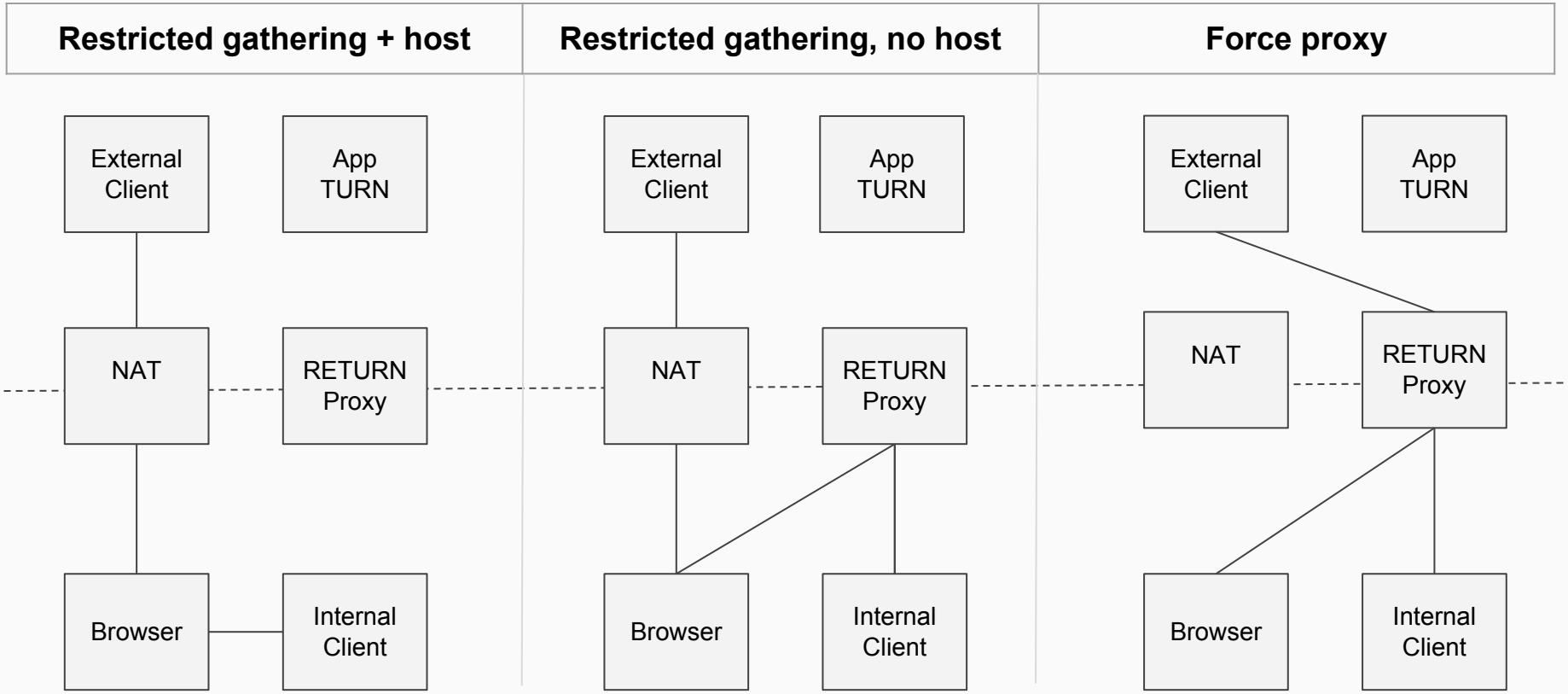
Proxies

- **Force proxy** mode was previously called "Force TCP and proxy"
- Adam argued that this was overly restrictive, as one could potentially have a UDP proxy (SOCKS5 or RETURN)
- Text changed to consider UDP proxies, and mention that the performance implications are not as dire in these (currently rare) cases
- One noteworthy difference with UDP proxies is that the proxy can be used even in Restricted Gathering scenarios (i.e. not just Force Proxy)

HTTP/SOCKS Proxy Behavior



RETURN Proxy Behavior



Open Issue: RFC 2119 Language

- Is this a standards-track doc?
- If standards-track, shouldn't it be more prescriptive?
- Harald suggests: **yes** and **yes**, e.g.:

Old:

"We recommend Mode 1 as the default behavior only if cam/mic permission has been granted, or Mode 2 if this is not the case."

New:

"WebRTC implementations MUST implement Modes 1 and 2. Mode 2 SHOULD be the default behavior, with Mode 1 only activated if the user has granted permission."

Next Steps

- Resolve wording issue and publish new version
- Additional reviews?
- WGLC?