ECRIT WG

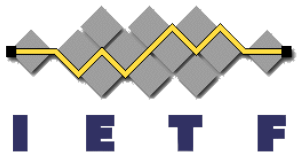
draft-aboba-rtcweb-ecrit-00

Bernard Aboba

Martin Thomson

July 30, 2012

IETF 84, Vancouver



Please join the Jabber room:
ecrit@jabber.ietf.org

Purpose of the Document

- To explore how emergency services functionality can be implemented within the WebRTC framework, including:
  - Location (Section 2)
  - Call routing (Section 2)
  - Accessibility (Section 4)
  - Interoperability with PSAPs implementing next generation emergency services
    - Media (Section 3)
- Document will evolve based on implementation experience.

Caveats

- The document provides no guidance as to whether a given WebRTC application or service will be subject to emergency service obligations.
  - See caveat in [PhoneBCP] Section 4
- The document does not advocate use of IP-based communications in all circumstances.
  - Where accurate location is unavailable (or the device does not have access to the Internet) alternatives may be preferable
    - Example: Could use WebAPI/WebTelephony API to access underlying (cellular) telephony platform.

Context

- Document starts from requirements in [PhoneBCP], while recognizing that:
  - RFC 6443 and [PhoneBCP] assume the use of SIP as the signaling mechanism for emergency calling.
  - Signaling is out of scope for WebRTC.
- Implications
  - SIP-related requirements do not necessarily apply to WebRTC implementations, applications and services.
  - Focus of the document is on requirements that are independent of the signaling mechanism.

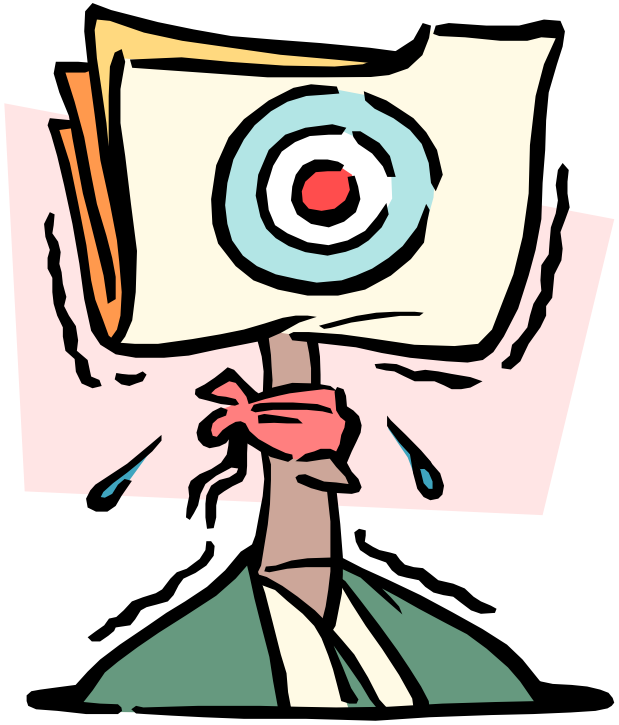
Location and Call Routing

- Automatically obtaining location suitable for emergency use is highly desirable for WebRTC emergency applications. Potential approaches:
  - Implementation in Javascript
    - Both HELD [RFC5985] and LoST [RFC5222] are implementable in JS
    - Biggest challenge is server location:
      - Mechanisms described in [RFC5986] and [RFC5223] based on DHCP option typically not retrievable by a browser application.
      - handled on server-side)
      - LoST server could be provided by the emergency services application.
  - GeoLocation API
    - Not developed with emergency uses in mind.
      - Example: source of the location information not provided, and can be difficult to infer.
    - Currently, underlying services disclaim applicability for emergency uses, and do not consistently provide the required accuracy.
  - WebAPI/WebTelephony API

Accessibility

- WebRTC-based emergency services SHOULD conform to the Web Content Accessibility Guidelines (WCAG) v2.0.
- Support for text communications
  - W3C developing proposed charter for the Timed Text Working Group, which may produce a second edition of TTML v1.0 as well as TTML v1.1.
  - [ED-75] Instant Messaging requirement can be met in Javascript
    - SIP MESSAGE [RFC3428]
    - XMPP [RFC6121]
  - [ED-76] requirement for Realtime Text (RFC 4103) not applicable to WebRTC
    - RFC 4103 typically implemented along with SIP signaling [RFC5194].
    - XEP-0301 implementable in JS with adequate performance, can support realtime text in both point-to-point and multi-user-chat [XEP-045] scenarios.

Media Requirements

- Silence suppression [ED-74], RTP [ED-71] requirements met by WebRTC implementations conforming to the RTP usage profile.
- Video: Disposition of [ED-77] requirement unclear:
  - In emergency services, requirements need not be symmetric on the browser and the PSAP. PSAP can support multiple codecs, and browser then only needs to support one of them to enable interoperability.
  - than attempting to drive mainstream acceptance based on regulations.
- Audio: [ED-73] does not require G.711 support if the service supports transcoding but G.711 support is desirable for PSTN interop, so recommend making G.711 mandatory-

Feedback?