WebRTC

WebRTC (**Web Real-Time Communication**) is a free, open-source project that provides web browsers and mobile applications with real-time communication (RTC) via simple application programming interfaces (APIs). It allows audio and video communication to work inside web pages by allowing direct peer-to-peer communication, eliminating the need to install plugins or download native apps.[2] Supported by Google, Microsoft, Mozilla, and Opera, WebRTC is being standardized through the World Wide Web Consortium (W3C) and the Internet Engineering Task Force (IETF).[3]

Its mission is to "enable rich, high-quality RTP applications to be developed for the browser, mobile platforms, and IoT devices, and allow them all to communicate via a common set of protocols".[3] The reference implementation is released as free software under the terms of a BSD license. OpenWebRTC provides another free implementation based on the multimedia framework GStreamer. JavaScript inventor Brendan Eich called it a "new front in the long war for an open and unencumbered web".[4]

WebRTC

Original author(s)	Justin Uberti Peter Thatcher
Initial release	2011
Stable release	1.1 / May 4, 2017
Repository	github.com/webrtc/apprtc
Written in	C++[1], Javascript
License	BSD license
Website	webrtc.org

Contents

History

In May 2010, Google bought Global IP Solutions or GIPS, a VoIP and videoconferencing software company that had developed many components required for RTC, such as codecs and echo cancellation techniques. Google open-sourced the GIPS technology and engaged with relevant standards bodies at the IETF and W3C to ensure industry consensus.[5][6] In May 2011, Google released an open-source project for browser-based real-time communication known as WebRTC.[7] This has been followed by ongoing work to standardize the relevant protocols in the IETF[8] and browser APIs in the W3C.[9]

In May 2011, Ericsson Labs built the first implementation of WebRTC using a modified WebKit library.[10] In October 2011, the W3C published its first draft for the spec.[11] WebRTC milestones include the first cross-browser video call (February 2013), first cross-browser data transfers (February 2014), and as of July 2014 Google Hangouts was "kind of" using WebRTC.[12]

The W3C draft of WebRTC[13] is a work in progress with advanced implementations in the Chrome and Firefox browsers. The API is based on preliminary work done in the WHATWG.[14] It was referred to as the ConnectionPeer API, and a pre-standards concept implementation was created at Ericsson Labs.[15] The WebRTC Working Group expects this specification to evolve significantly based on:

- Outcomes of ongoing exchanges in the companion RTCWEB group at IETF[16] to define the set of protocols that, together with this document, define real-time communications in web browsers. While no one signaling protocol is mandated, SIP over Websockets (RFC 7118) is often used partially due to the applicability of SIP to most of the envisaged communication scenarios as well as the availability of open-source software such as JsSIP.
- Privacy issues that arise when exposing local capabilities and local streams
- Technical discussions within the group, on implementing data channels in particular[17]
- Experience gained through early experimentation
- Feedback from other groups and individuals

Overview

Design

Major components of WebRTC include several JavaScript APIs:

- `getUserMedia` acquires the audio and video media (e.g., by accessing a device's camera and microphone).[18]
- `RTCPeerConnection` enables audio and video communication between peers. It performs signal processing, codec handling, peer-to-peer communication, security, and bandwidth management.[19]
- `RTCDataChannel` allows bidirectional communication of arbitrary data between peers. It uses the same API as WebSockets and has very low latency.[20]

The WebRTC API also includes a statistics function:

- `getStats` allows the web application to retrieve a set of statistics about WebRTC sessions. These statistics data are being described in a separate W3C document.[21]

RFC 7874 requires implementations to provide PCMA/PCMU (RFC 3551), Telephone Event as DTMF (RFC 4733), and Opus (RFC 6716) audio codecs as minimum capabilities. The PeerConnection, data channel and media capture browser APIs are detailed in the W3C.

W3C is developing ORTC (Object Real-Time Communications) for WebRTC.[22] This is commonly referred to as WebRTC 1.1.

Examples

Although initially developed for web browsers, WebRTC has applications for non-browser devices, including mobile platforms and IoT devices. Examples include browser-based VoIP telephony, also called cloud phones or web phones, which allow calls to be made and received from within a web browser, replacing the requirement to download and install a softphone.[23]

Support

WebRTC is supported by the following browsers:

- Desktop PC
  - Microsoft Edge 12+[24]
  - Google Chrome 28+
  - Mozilla Firefox 22+[25]
  - Safari 11+[26]
  - Opera 18+[27]
  - Vivaldi 1.9+

- Android
  - Google Chrome 28+ (enabled by default since 29)
  - Mozilla Firefox 24+[28]
  - Opera Mobile 12+
- Chrome OS
- Firefox OS
- Blackberry 10
- iOS 11
  - MobileSafari/WebKit
- Tizen 3.0

Support was not included in Internet Explorer prior to its final feature release in October 2013[29], but 3rd party plugins are available to add the support of WebRTC to IE and Safari for MacOS.[30][31] At WWDC 2017, Apple announced Safari would get WebRTC support in Safari 11,[26] and it was made available in release 32 of the Safari Technology Preview.[32]

Concerns

In January 2015, TorrentFreak reported a serious security flaw in browsers that support WebRTC, saying that it compromised the security of VPN tunnels, by exposing the true IP address of a user.[33] The IP address read requests are not visible in the browser's developer console, and they are not blocked by most ad blocking/privacy/security add-ons, enabling online tracking by advertisers and other entities despite precautions[34] (however the uBlock Origin add-on can fix this problem).[35]

See also

- Global IP Solutions (GIPS)
- Real-time Transport Protocol (RTP)
- Session Description Protocol (SDP)
- WebRTC Gateway

References

1. https://webrtc.googlesource.com/src/webrtc/
2. How WebRTC Is Revolutionizing Telephony (http://blogs.trilogy-lte.com/post/77427158750/how-webrtc-is-revolutionizing-telephony). Blogs.trilogy-lte.com (2014-02-21). Retrieved on 2014-04-11.
3. "WebRTC" (https://archive.is/svdbj). *WebRTC*. Archived from the original (https://webrtc.org) on 8 January 2018. Retrieved 6 February 2018.
4. Justin Uberti; Sam Dutton. "WebRTC: Plugin-free realtime communication" (https://io13webrtc.appspot.com/#6). Retrieved 6 February 2018.
5. "Are the WebRTC components from Google's acquisition of Global IP Solutions?" (https://webrtc.org/faq/#are-the-webrtc-components-from-googles-acquisition-of-global-ip-solutions). *WebRTC*. Retrieved 6 February 2018.
6. Wauters, Robin (18 May 2010). "Google makes $68.2 million cash offer for Global IP Solutions" (https://techcrunch.com/2010/05/18/google-makes-68-2-million-cash-offer-for-global-ip-solutions). *TechCrunch*. Retrieved 6 February 2018.
7. Harald Alvestrand (2011-05-31). "Google release of WebRTC source code" (http://lists.w3.org/Archives/Public/public-webrtc/2011May/0022.html). public-webrtc@w3.org. Retrieved 2012-09-12.
8. Charter of the Real-Time Communication in WEB-browsers (rtcweb) working group (http://datatracker.ietf.org/wg/rtcweb/charter/)
9. "WebRTC 1.0: Real-time Communication Between Browsers" (http://www.w3.org/TR/webrtc/). W3.org. Retrieved 2012-09-12.
10. Stefan Håkansson; Stefan Ålund (26 May 2011). "Beyond HTML5: Experiment with Real-Time Communication in a Browser" (https://www.ericsson.com/research-blog/beyond-html5-experiment-real-time-communication-browser). *Ericsson Research blog*. Retrieved 6 February 2018.

11. "WebRTC 1.0: Real-time Communication Between Browsers" (https://www.w3.org/TR/2011/WD-webrtc-20111027/). *World Wide Web Consortium*. 27 October 2011. Retrieved 6 February 2018.
12. Nowak, Szymon. "WebRTC: So Much More Than Videoconferencing" (https://szimek.github.io/presentation-meetjs-summit-2014-webrtc/#16). *GitHub*. Retrieved 6 February 2018.
13. "WebRTC 1.0: Real-time Communication Between Browsers" (http://dev.w3.org/2011/webrtc/editor/webrtc.html). Dev.w3.org. Retrieved 2012-09-12.
14. "Introduction — HTML Standard" (http://www.whatwg.org/specs/web-apps/current-work/multipage/introduction.html#history-1). Whatwg.org. Retrieved 2012-09-12.
15. "Beyond HTML5: Peer-to-Peer Conversational Video" (http://www.ericsson.com/research-blog/context-aware-communication/beyond-html5-peer-peer-conversational-video/). Labs.ericsson.com. Retrieved 2012-09-12.
16. "Rtcweb Status Pages" (http://tools.ietf.org/wg/rtcweb/). Tools.ietf.org. Retrieved 2012-09-12.
17. "draft-jesup-rtcweb-data-protocol-00 - WebRTC Data Channel Protocol" (http://tools.ietf.org/html/draft-jesup-rtcweb-data-protocol-00). Tools.ietf.org. Retrieved 2012-09-12.
18. "Media Capture and Streams: getUserMedia" (http://www.w3.org/TR/mediacapture-streams/#dom-navigator-getusermedia). W3C. 2013-09-03. Retrieved 2014-01-15.
19. "WebRTC: RTCPeerConnection Interface" (http://www.w3.org/TR/webrtc/#rtcpeerconnection-interface). W3C. 2013-09-10. Retrieved 2014-01-15.
20. "WebRTC: RTCDataChannel" (http://www.w3.org/TR/webrtc/#rtcdatachannel). W3C. 2013-09-10. Retrieved 2014-01-15.
21. "Identifiers for WebRTC's Statistics API" (https://w3c.github.io/webrtc-stats/webrtc-stats.html). W3C. 2014-09-29.
22. "W3C ORTC (Object Real-time Communications) Community Group" (http://www.w3.org/community/ortc/).
23. https://www.siliconrepublic.com/comms/babelfish-softphone-cloud-goldfish
24. "ORTC API is now available in Microsoft Edge" (https://blogs.windows.com/msedgedev/2015/09/18/ortc-api-is-now-available-in-microsoft-edge/). Microsoft.
25. Firefox Notes - Desktop (https://www.mozilla.org/en-US/firefox/22.0/releasenotes/). Mozilla.org (2013-06-25). Retrieved on 2014-04-11.
26. "Safari 11.0" (https://developer.apple.com/library/content/releasenotes/General/WhatsNewInSafari/Safari_11_0/Safari_11_0.html). Apple Inc. Retrieved 6 June 2017.
27. Opera News (http://blogs.opera.com/news/2013/11/opera-18/). blogs.opera.com (2013-11-19). Retrieved on 2015-09-17.
28. Firefox Notes - Desktop (https://www.mozilla.org/en-US/mobile/24.0/releasenotes/). Mozilla.org (2013-09-17). Retrieved on 2014-08-04.
29. "Internet Explorer Web Platform Status and Roadmap" (https://web.archive.org/web/20150504092553/https://status.modern.ie/webrtcwebrtcv10api?term=WebRTC). Microsoft. Archived from the original (https://status.modern.ie/webrtcwebrtcv10api?term=WebRTC) on 2015-05-04. Retrieved 7 September 2014.
30. "Priologic Releases First Open Source WebRTC Plugin for Internet Explorer" (https://web.archive.org/web/20140626033329/http://easyrtc.com/news/?a=37af688dfd72da30a2ccfd839a9a3d24). 2014-06-10. Archived from the original (http://www.easyrtc.com/news/?a=37af688dfd72da30a2ccfd839a9a3d24) on 2014-06-26.
31. Wilcox, Charles (2014-05-12). "Temasys Plugin Supports webRTC in Internet Explorer and Apple Safari, on Desktops" (http://www.prweb.com/releases/2014/05/prweb11844308.htm) (Press release). Temasys Communications Pte Ltd. PRWeb.
32. "Safari Technology Preview Release Notes" (https://developer.apple.com/safari/technology-preview/release-notes/). Retrieved 12 June 2017.
33. Huge Security Flaw Leaks VPN Users' Real IP-addresses (https://torrentfreak.com/huge-security-flaw-leaks-vpn-users-real-ip-addresses-150130/) TorrentFreak.com (2015-01-30). Retrieved on 2015-02-21.
34. STUN IP Address requests for WebRTC (https://github.com/diafygi/webrtc-ips) Retrieved on 2015-02-21.
35. Raymond Hill (26 March 2016). "Prevent WebRTC from leaking local IP address" (https://github.com/gorhill/uBlock/wiki/Prevent-WebRTC-from-leaking-local-IP-address). *uBlock Origin documentation*. Retrieved 1 Sep 2016.

Further reading

- Proust, S., ed. (May 2016). *Additional WebRTC Audio Codecs for Interoperability*. IETF. doi:10.17487/RFC7875. RFC 7875. https://tools.ietf.org/html/rfc7875. Retrieved 2016-10-12.
- Valin, J. M.; Bran, C. (May 2016). *WebRTC Audio Codec and Processing Requirements*. IETF. doi:10.17487/RFC7874. RFC 7874. https://tools.ietf.org/html/rfc7874. Retrieved 2016-10-12.
- Roach, A. B. (March 2016). *WebRTC Video Processing and Codec Requirements*. IETF. doi:10.17487/RFC7742. RFC 7742. https://tools.ietf.org/html/rfc7742. Retrieved 2016-10-12.
- Perumal, M.; Wing, D.; Ravindranath, R.; Reddy, T.; Thomson, M. (October 2015). *Session Traversal Utilities for NAT (STUN) Usage for Consent Freshness*. IETF. doi:10.17487/RFC7675. RFC 7675. https://tools.ietf.org/html/rfc7675. Retrieved 2016-10-12.
- Holmberg, C.; Hakansson, S.; Eriksson, G. (March 2015). *Web Real-Time Communication Use Cases and Requirements*. IETF. doi:10.17487/RFC7478. RFC 7478. https://tools.ietf.org/html/rfc7478. Retrieved 2016-10-12.

External links

- Official website
- W3C Web Real-Time Communications Working Group
- IETF Real-Time Communication in WEB-browsers (rtcweb) Working Group
- Getting Started With WebRTC
- WebRTC Interest and History
- WebRTC:How and Why?
- WebRTC: IP Address Exposure Demo