CARENA: Delivering Enterprise-Class Communications with WebRTC

A Carena White Paper

WebRTC, an emerging web standard, has been a game changer for Carena's telehealth communications. By simplifying the process of adding voice and video to any web browser, WebRTC can turn any device with a browser into a phone and make communications from inside web apps both ubiquitous and cost-efficient.

CARENA

TABLE OF CONTENTS

REFERENCES

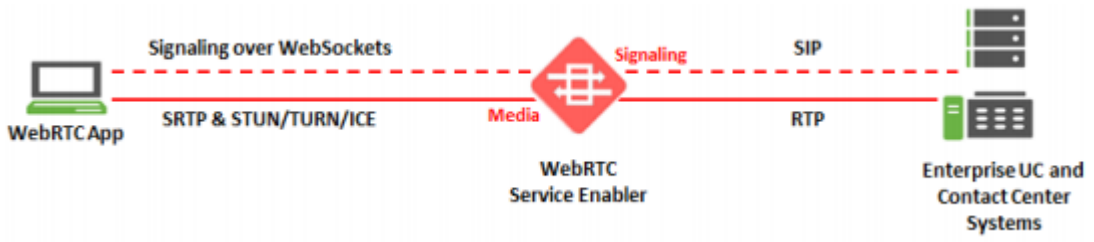
http://www.onsip.com/webrtc-sip-network/webrtcimplementation/webrtc-security

http://www.pubnub.com/blog/whatiswebrtc/

http://www.hhs.gov/ocr/privacy/hipaa/administrative/securityrule/security101.pdf

What is WebRTC?

WebRTC (Web Real-time Communications) enables peer to peer video, audio, and data communication between two web browsers. This allows for video calling, video chat, and peer to peer file sharing entirely in the web browser, with no plugins.



How does Carena work with WebRTC?

The why

Because WebRTC requires no plugins, frameworks or applications, all you need is a WebRTC compatible browser. To an end user, WebRTC applications “just work” right out of the box. No Flash, no Silverlight, no JavaScript API, just pure video, audio, and data communication on any webpage.

The how

WebRTC allows Carena to provide secure real time audio, video and data sharing capability to patients through our custom built web portals.

WebRTC and compliance with the HIPAA act

HIPAA

As defined by HIPAA, there are two types of entities that must comply with the rules: the covered entities and the business associates. Covered entities are the health care providers and insurances companies that maintain Protected Health Information. Business associates are a person or organization that conducts business with the covered entity that involves the use or disclosure of individually identifiable health information. In other words, telehealth apps that access PHI are considered as business associates, and so they must be HIPAA compliant.

CARENA

Compliance

Carena has architected their applications in a manner to be compliant with the HIPAA Privacy and Security Rules. In the case of a video consultation, Carena only transmits the encrypted PHI and at no point has access to the encryption key. By encrypting media streams, WebRTC addresses these requirements. Carena does not store video or chat sessions.

For real-time digital communication of patient information, HIPAA requires that the communication channel be properly secured to protect patient confidentiality. Carena ensures secure transmission by using the following implementations.

Signaling: Carena's Homegrown signaling layer is encrypted in addition to the media. The signaling server uses Secure WebSockets (wss:// instead of ws://), which uses TLS to encrypt the WebSocket connection.

Secure Connection: The sessions established are secure (with secured tokens that are regenerated). Random AES keys are generated by clients at the beginning of the media connection and, to increase security, additional keys are generated periodically throughout the session.

Data Transmission and Encryption: Carena employs Transport Layer Security (TLS) to encrypt both voice and video data.

The core protocols used are SRTP for media traffic encryption and DTLS-SRTP for key negotiation, both of which are defined by the IETF.

The endpoints use AES cipher with 128-bit keys to encrypt audio and video, and HMAC-SHA1 to verify data integrity.

Turn Server: A direct peer to peer connection is attempted first and, only if it fails (likely due to blocked ports or network topology), a connection is attempted through a TURN server. The TURN server (located inside of Carena's Datacenter) simply relays packets from one peer to the other - it does NOT decrypt anything.

WebRTC Security by OnSip.com

WebRTC has built-in features that address quite a few security concerns. These features are not optional unlike other video platforms. Malware may come packaged with plugin downloads, but users of WebRTC applications do not need to download any plugins. The underlying components that allow real time communications run in the browser's sandbox, and are updated automatically when the browser updates. With WebRTC applications, the end user must grant explicit permission before the browser is allowed access to his/her local devices. Furthermore, when the camera and/or microphone are active, the browser will display an 'active' indicator, usually found in the browser tab. WebRTC security measures ensure that media is automatically encrypted.

In order for WebRTC to transmit real-time data (webcam, microphone, text), it must first encrypt this information using the Datagram Transport Layer Security (DTLS) method (defined by RFCs 6347, 5238, 6083, 5764). DTLS is a standardized protocol built into all browsers that support WebRTC. It is a protocol designed to prevent eavesdropping and information tampering. The method was modeled on Transport Layer Security (TLS), a protocol that offers full encryption with asymmetric cryptography methods, data confidentiality, and message authentication. It is consistently used in web browsers, email, and VoIP platforms to encrypt information.

This ensures that WebRTC data can be secured via any standard SSL based connection on the web. WebRTC security offers end-to-end encryption between peers with almost any server arrangement. For instance, a TURN server will only parse the UDP layer of a WebRTC packet. It cannot understand or modify the application data layer (the real-time WebRTC data). In other words, servers will not decode the sensitive information peers send to each other in order to route it.

The signaling layer can be encrypted in addition to the media. The mechanism obviously depends on the signaling layer chosen.

A signaling layer can provide an authentication and authorization mechanism to determine the identity of the user. For example, Carena uses usernames and passwords to ensure that you are who you say you are before allowing you to place a video call.