

Design and implementation of a novel secured and wide WebRTC signalling mechanism for multimedia over internet

Naktal Edan¹, Enas Y. Abdullah²

¹College of Computers Sciences and Mathematics, Mosul University, Iraq

²Department of Mathematics, Education College for Girls, University of Kufa, Iraq

Article Info

Article history:

Received Dec 25, 2019

Revised Mar 27, 2020

Accepted Apr 11, 2020

Keywords:

4 generation (4G)

ASP.net & C#

Internet

Mesh topology

Quality of experience (QoE)

Web real-time communication

(WebRTC)

ABSTRACT

A modern and free technology called web real-time communication (WebRTC) was enhanced to allow browser-to-browser multimedia communication without plugins. In contract, WebRTC has not categorised a specific signalling mechanism to set, establish and end communication between browsers. The primary target of this application is to produce and implement a novel WebRTC signalling mechanism for multimedia communication between different users over the Internet without plugins. Furthermore, it has been applied over different browsers, such as Explorer, Safari, Google Chrome, Firefox and Opera without any downloading or fees. This application designed using JavaScript language under ASP.net and C# language. Moreover, to prevent irrelevant users from accessing or attacking the session, user-id for initiating and joining the course using encryption technique was done. This system has been employed in real implementation among various users; therefore, an evaluation of bandwidth consumption, CPU, and quality of experience (QoE) was accomplished. The results show an original signalling mechanism which applied to different browsers, multiple users, and diverse networks such as Ethernet and Wireless. Besides, it sets session initiator, saves the communication efficient even if the initiator leaves, and communicating new participator with existing participants, etc. This studying focuses on the creation of a new signalling mechanism, the limitations of resources for WebRTC video conferencing.

*Copyright © 2020 Institute of Advanced Engineering and Science.
All rights reserved.*

Corresponding Author:

Naktal Edan,

College of Computers Sciences and Mathematics,

Mosul University,

Mosul, Iraq.

Email: edannaktal@gmail.com, naktal.edan@uomosul.edu.iq

1. INTRODUCTION

Real-time multimedia communication over the Internet has been supported by communication technology in different directions [1]. Also, using the Internet to deliver multimedia becomes a more appropriate selection as long as it offers better quality services and low cost [2-4]. Last years, web real-time communication (WebRTC) had been established by the Internet Engineering Task Force (IETF) and World Wide Web consortium (W3C) as an original standard [5]. WebRTC is a set of open-source and a collection of JavaScript libraries [6] that offers collaborative communications of audio and video [7]. Indeed, using WebRTC technology no need for plug-ins, high reliability, free and no licensing [6, 8]. On contrary, W3C and IETF have not determined a particular WebRTC signalling mechanism to execute it [9-12]. The reason behind that is signalling left to the developer to select from the current protocols, for example, Session initiation protocol (SIP) or extensible messaging and presence protocol (XMPP) or to produce their contract [13, 14]. Importantly, to maximise the compatibility with established technologies [15, 16], or for a novel use case.

Signalling is significant to support creating communication between browsers by swapping data via channels [5]. Furthermore, signalling supports browser to connect the server and enables a connection between participants with the server [17]. Figure 1 displays the signalling architecture in WebRTC.

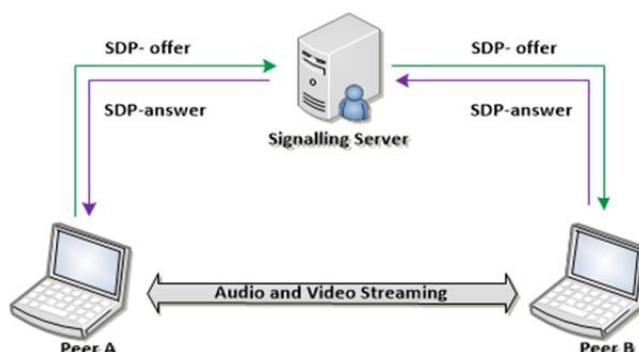


Figure 1. A general signalling mechanism

Consequently, this research is concentrated on different challenged as following: (a) WebRTC signalling mechanism matters left on the researchers, so researchers should create it, (b) using the same server to provide scalability for many peers, (c) selecting the notable browser is still undefined, such as Chrome, Firefox or Opera, (d) using "user-id" for joining the same room, and make sure that irrelevant messages will not be shared with unknown users, and (e) most of the current WebRTC signalling mechanism or protocols have various restrictions; for example, it is complicated to provide bi-directional communication, using for finite users and so on. This research aims to generate the following:

- A new WebRTC signalling mechanism for video communication in a real-world application that enables bidirectional conferencing.
- Provide an extensive connection through Ethernet and Wireless of the Internet.
- Using a secured mechanism by generating "user-id" for joining the same and/or different rooms.
- Applying in Google Chrome, Firefox, Opera, Explorer, and Safari.
- Evaluate resources, such as CPU capabilities, bandwidth speed, Quality of Experience (QoE) through actual users, and different computer systems.
- Do not use external software (plugins), hardware, clouds, or commercial servers. In contrast, it has been used JavaScript language under ASP.net and C# language.
- It sets session initiator, saves the communication efficient even if the initiator leaves, and communicating new participator with existing participants, etc.

Accordingly, this application produces and implements a novel WebRTC signalling mechanism for video conferencing between dissimilar users over the Internet without plugins. Additionally, it has been applied over Explorer, Safari, Google Chrome, Firefox and Opera with no fees. Besides, it prevents irrelevant users from attacking the chat room by creating (user-id) using an encryption technique. Off-course, this system has been employed in a real implementation, and the outcomes show a creative signalling mechanism. In contrast, it has been applied to different browsers, multiple users, and diverse networks. This work is arranged as shows; Section 2 focuses on WebRTC related work. In section 3 describes the methodology, test and analysis of this research. Section 4 deliberates the evaluation. Finally, conclusion and future tasks have explained in Section 5.

2. RELATED WORK

Some applications have utilised XMLHttpRequest (XHR), but it does not support bi-directional communication approach [18]. Likewise, in [19] illustrated that XHR more complicated than other protocols. Also, [20] explained that all features of existing browser have not supported by XHR. Additionally, in [15, 16, 21] indicated that SIP clients are not yet embedded within existing protocols that needed for WebRTC, including developing SIP clients given less speed and cost more than the existing real-time applications. Based on the XMPP, [22] recommended that using XMPP has various limitations on multiple web pages, the retest is needed, and is a slow protocol which leads to massive circulation through the internet [23]. By contrast, using Jingle can only support a session for two peers and shows latency in the server response [24].

Signalling mechanism in WebRTC has considered as the first part of the application, which has not been defined yet [25]. Similarly, it has been confirmed in [26] that signalling is the core distinguished implementation issue on WebRTC. This issue was considered as the first one to be faced when executing WebRTC. Thus, the existing version of the WebRTC is not mainly endorsed “multi-browser” communication over browsers [5, 7, 27, 28]. Hence, WebRTC requires a signalling mechanism or protocol to establish a connection between different peers [22, 29, 30].

3. METHODOLOGY, IMPLEMENTATION, AND ANALYSIS

3.1. Methodology

This implementation has been applied to different locations, browsers and devices to provide bidirectional video conferencing in the real world. Therefore, it has been conducted a session for training, meetings, and presentations through the Internet between many users. Many different techniques were used to create the signalling mechanism of this work, such as (a) JavaScript language for useful content of video conferencing through HTML5, (b) Secure socket layer (SSL) to encrypt connections between users, and prevent undefined visitor to enter the session; also to control the meeting by the initiator, (c) ASP.net to build a dynamic HTML page and guarantee the reliability, and flexibility, (d) C# as a core programming language for designing, and (e) SQL hash table for participants database in order to save and organise user’s data. Equally, two concepts have been considered to build the signalling mechanism like offer and answer. Mainly, to establish a session, there should be a room creator to adjust "user-id" for other participants. All peers must always utilise the identical "user-id" to go into the session and communicates participants. Furthermore, using "getLocalDescription" method (to access camera and microphone), "createOffer" and "createAnswer" methods (to exchange Session Description Protocol (SDP) that holds information of bandwidth, video and audio codecs, etc.), and "createDataChannel" method (to transfer data through opening a channel). Accordingly, Peer A can initiate a session, and when peer B would like to join the room, it sends a request to peer A for the accessibility. In contrast, once peer A received the request, it validates the "user-id" and confirms the availability with the SDP. Now both peers A&B can exchange the cryptographic parameters based on datagram transport layer security (DTLS) and secure real-time transport protocol (SRTP); as a result, they both arrange (RTC) data transported. This signalling mechanism has a novel target to permit various users to enter specific or different sessions using bi-directional video conferencing. Regarding the analysis, different software was utilised to evaluate the resources as showing: (a) Wireshark analyser to discover bandwidth consumption, and (b) Task Manager to show a CPU performance.

3.2. Implementation and results

The system has been tested in the real world over both network and Internet, so the result shows that it has been worked seamlessly and the quality of video streaming was relying on bandwidth consumption and CPU performance. In this application, the primary mechanism has been created based on ASP.net website, which programmed using C# and JavaScript languages as a client-server system. In other words, this system contains two parts, such as a video streaming server and client device. Figure 2, presents the structure and operation of the created order.

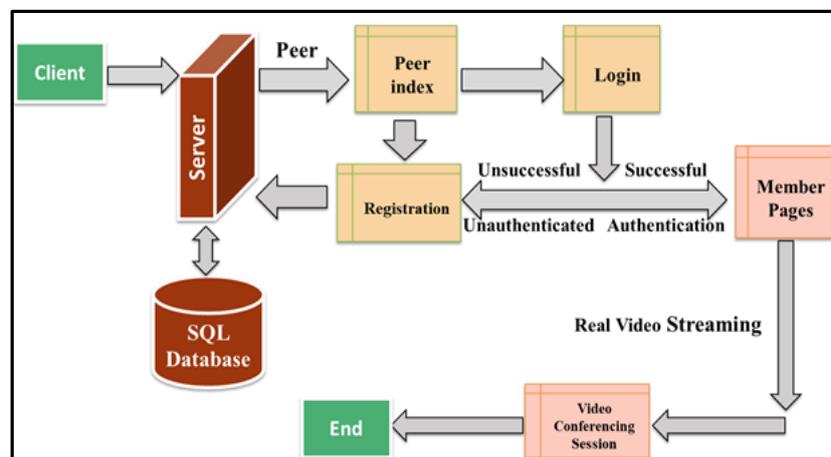


Figure 2. Construction of the designed system

The principal webpage has been uploaded on ASP host server to achieve the implementation in real-time. Regarding the security, the process presents that it is impossible to access the main webpage before entering the specific user-id. In other words, the "user-id" has encrypted in the client-side, which is not allowed to be known; so this system offers privacy for every user. Figure 3, shows the main webpage of this system, and the Table 1 presents password value after encryption. Also, Table 2 shoes SQL database registration information.

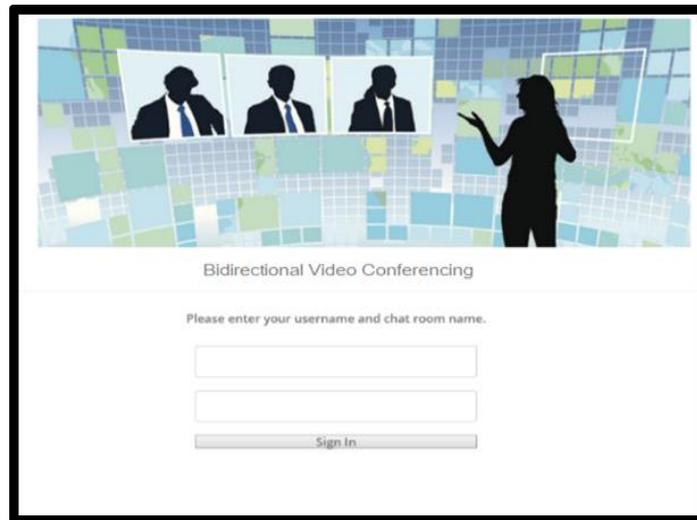


Figure 3. Videoconferencing main webpage

Table 1. Password value after encryption

Peer's name	Password key	Hash Value
Ali	123	upzCzAYKaIwFDWG++IBJ78ICTKsmLNcaRhvCrKGGH8/Bn3LHbA7yBuv8 WzOro4hrz8wzqEka86EdI/+Yx02cXoVg==
John	123	oypxDbAYKaIwFDWG++IBJ22ICTKsmLNcaRhvCrKGGH8/Bn2LHbA7yBuv8 WzOro3hrz8wzqEka86EdI/+Yx02cXpww==
Scott	123	aaxuQcAYKaIwFDWG++IBJ99ICTKsmLNcaRhvCrKGGH8/Bn1LHbA7yBuv8 WzOro2hrz8wzqEka86EdI/+Yx02cXapw==

Table 2. Database registration information

User name	Pass
Ali	AYM237oD1BaBPKSTDL6LxS34dHc4LS/w8fCSxv4g2Scvbn...
John	ASfO3O/CGvNHDeKbCDF5vbXvksks/sqww33HJNHJNDVerTh...
Scott	AS+ddVcGHju4rtygf+asdo29fd2c/dkngkdsFG1fgT+dfjRRTGYM..

The communication of video conferencing has executed using different operating systems and browsers. Due to this, it has tested used windows 7, 8 and 10, MAC operating system, and smartphones (Android and iPhone). As well as, it has considered the ability of various browsers, such as Internet Explorer, Google Chrome, Firefox, and Opera, as shown in Table 2. From Table 3, the outcomes display that not all browsers can support WebRTC video conferencing due to some browsers is not able to operate JavaScript. In contrast, the test emphasised that the best communication was using Firefox, Chrome, and Opera.

Table 3. Results of applying the designed mechanism on different browsers, systems, and devices

Browsers	iPhone	Android	MAC computer	Computer		
Operating System	OX 10	Android 7	OSX	Windows 7	Windows 8	Windows 10
Google Chrome	No	Yes	Yes	Yes	Yes	Yes
Firefox	No	Yes	Yes	Yes	Yes	Yes
Opera	No	Yes	Yes	Yes	Yes	Yes
Explorer	No	No	No	No	No	No
Safari	No	No	No	No	No	No

3.2.1. Quality and video conferencing

The evaluation of audio and video quality was done via individual test between users over the Internet networks; thus, the quality was excellent. However, due to smartphone capabilities, bandwidth speed and low CPU performance, some quality was disabled when the third participant joined the room using a smartphone; and also when the fifth participant joined the room using CPU core i3&i5. In [31, 32] emphasised that the CPU core affects the number of peers and bandwidth consumption impacts on the quality of multimedia communication. To put it differently, [33] confirmed that CPU requests high processor to decode or encode the multimedia (audio and video), also bandwidth consumption can limit the quality of multimedia streaming [34]. As a result, in this research, the bandwidth measurement presented that each user consumes 2 Mb/s.

3.2.2. Quality of experience

In this part, five users engaged in this test to grant their thoughts individually on the comprehended participant expertise via the usage of questionnaires as clarified in Table 4. According to assessment, it can consider the bandwidth consumption as a foundation of reducing the quality of video; and also the CPU performance as a cause of controlling the number of users.

Table 4. QoE based on five users

Questions	Very Bad Very annoying	Bad Annoying	Fair Slightly annoying	Good Perceptible but not annoying	Excellent Imperceptible
Estimate the ease of using the application					5
Estimate the quality of audio during the session				2	3
Estimate the quality of the video during the session				1	4
Estimate the resilience of the connection				1	4
Estimate the echo during the session				2	3

4. CONCLUSION AND FUTURE EFFORT

In this paper, it has proved that a new signalling mechanism has been designed and applied. Also, this application can be used to achieve video conferencing among different browsers (Chrome, Firefox, and Opera), different devices (computers, and mobiles), and various operating systems (Windows 7, 8, and 10). The test of this research was using a new signalling mechanism to set up, established and closes a bidirectional video communication over the Internet. Besides, it offers a visual demo with users that need depth clarification and facing interaction. In the same way, it can keep the session productive even if any user leaves, and specify user-id to protect the course from unknown users. This application can be considered as the unique project that achieved a WebRTC bidirectional video conversation utilising the Internet network with no need for any external software. Besides, CPU performance and bandwidth speed have core problems in audio and video communications, Last but not least, the QoE confirmed that the designed application runs appropriately, and that can be led to produce more tests in the future.

REFERENCES

- [1] A. A. Lozano, "Performance analysis of topologies for Web-based Real-Time Communication (WebRTC)," Aalto University, 2013.
- [2] P. Biggs, "The Status of Voice Over Internet Protocol (VOIP) Worldwide," *International Telecommunication Union: The Future of Voice*, Geneva, 2007.
- [3] S. M. R. Sajjad and M. N. Dilber, "Comparative analysis of traditional telephone and VoIP systems," *Journal of Independent Studies and Research – Computing*, vol. 12, no. 1, pp. 25-31, 2014.
- [4] M. A. A. Suliman and A. Babiker A., "UMTS VoIP Codec QoS Evaluation," *IOSR Journal of Electronics and Communication Engineering*, vol. 10, no. 2, pp. 7-12, 2015.
- [5] J. J. Jaccard, et al., "WebRTC-Based Video Conferencing Service for Telehealth," *Computing*, vol. 98, no. 1, pp. 169-193, 2016.
- [6] M. Phankokkruad and P. Jaturawat, "An Evaluation of Technical Study and Performance for Real-Time Face Detection Using Web Real-Time Communication," in *International Conference on Computer, Communication, and Control Technology (I4CT)*, pp. 162-166, 2015.
- [7] G. Carullo, et al., "A Performance Evaluation of WebRTC over LTE," in *12th Annual Conference on Wireless On-demand Network Systems and Services (WONS)*, pp. 170-175, 2016.
- [8] E. Fosser and L. Nedberg, "Quality of Experience of WebRTC based video communication," Norwegian University of Science and Technology, 2016.

- [9] C. Cola and H. Valean, "On multi-user web conference using WebRTC," in *18th International Conference on System Theory, Control and Computing (ICSTCC)*, pp. 430-433, 2014.
- [10] A. Johnston, et al., "Taking on webRTC in an enterprise," *IEEE Communications Magazine*, vol. 51, no. 4, pp. 48-54, 2013.
- [11] S. Johansson, "Behaviour of WebRTC in Non-optimal Networks," Luleå University of Technology, 2018.
- [12] A. P. González, "Definition of a Mena Opinion Score for VP8 Over Real-Time Connections," Universida de Vigo, 2017.
- [13] V. Singh, et al., "Performance analysis of receive-side real-time congestion control for WebRTC," in *20th International Packet Video Workshop*, pp. 1-8, 2013.
- [14] P. J. Ha and L. D. Hoon, "Scalable signalling protocol for Web real-time communication based on a distributed hash table," *Computer Communications*, vol. 70, pp. 28-39, 2015.
- [15] B. Sredojev, et al., "WebRTC technology overview and signalling solution design and implementation," in *38th International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO*, pp. 1006-1009, 2015.
- [16] GSMA, "WebRTC to complement IP Communication Services version 1.0," GSM Associaton, 2016.
- [17] S. Rajab, "Comparing different network topologies for WebRTC conferencing," KTH Royal Institute of Technology, 2015.
- [18] R. Rai, "Socket. IO Real-time Web Application Development," *PACKT*, 2013.
- [19] P. Smolka, "Real-time Communication in Web Browser," Masaryk University, 2013.
- [20] M. V. D. Khoa, et al., "HTML5-based Distributed and Interactive E-learning Framework," 한국통신학회 종합 학술 발표회 논문집 (하계) - Proceedings of the Korean Society of Communication Sciences Conference (Summer), Korean, 2014.
- [21] M. Deshpande and S. P. Mohani, "Integration of WebRTC with SIP – Current Trends," *International Journal of Innovations in Engineering and Technology*, vol. 6, no. 2, pp. 92-96, 2015.
- [22] H. Rocha and R. L. Pereira, "Hyper-linked Communications: WebRTC enabled asynchronous collaboration," *IEEE International Conference on Communications*, pp. 1-7, 2017.
- [23] C. Fan, "Research on Development and Evaluation of WebRTC Signaling based on XMPP," Norwegian University of Science and Technology, 2017.
- [24] A. El Hamzaoui, et al., "A Formal Model for WebRTC Signaling Using SDL," *Network Systems: 4th International Conference NETYS 2016*, pp. 202-208, 2016.
- [25] M. Schindler, et al., "Evaluating Framework for Monitoring and Analyzing WebRTC Peer-to-Peer Applications," in *Proceedings of the Eleventh International Network Conference (INC)*, pp. 171-175, 2016.
- [26] A. Amirante, et al., "On the seamless interaction between webRTC browsers and SIP-based conferencing systems," *IEEE Communications Magazine*, vol. 51, no. 4, pp. 42-47, 2013.
- [27] C. Chiang, et al., "A video conferencing system based on WebRTC for seniors," in *International Conference on Trustworthy Systems and their Applications*, Taichung, pp. 51-56, 2014.
- [28] Y. Chen, et al., "WEBRTC on Multi-Party Communication to Lower Video Streaming Traffic," *International Journal of Advances in Electronics and Computer Science*, vol. 3, no. 2, pp. 46-49, 2016.
- [29] J. Uberti, et al., "Javascript Session Establishment Protocol," in *Network Working Group*, Internet-Draft, 2016.
- [30] J. C. Zhang, et al., "Getting Started with WebRTC and Test Driven Development," Medium, 2016.
- [31] N. M. Edan, et al., "WebNSM: A Novel WebRTC Signalling Mechanism for One-to-Many Bi-directional Video Conferencing," in *Computing Conference*, pp. 1-6, 2018.
- [32] N. Edan, et al., "Desing and Implement A Hybrid WebRTC Signalling Mechanism for Unidirectional & Bi-directional Video Conferencing," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 8, no. 1, pp. 390-399, 2018.
- [33] J. K. Nurminen, et al., "P2P media streaming with HTML5 and WebRTC," in *IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 63-64, 2013.
- [34] M. Pasha, et al., "Analysis of challenges faced by WebRTC videoconferencing and a remedial architecture," *International Journal of Computer Science and Information Security*, vol. 14, no. 10, pp. 698-705, 2016.