

Performance Analysis of WebRTC and SIP for Video Conferencing

Navrattan Parmar and Virender Ranga

Abstract: With the advancement in communication and development of technologies like VoIP and Video Conferencing, Web Real-Time Communication (WebRTC) is developed to communicate without plugins and stream the videos on a real time. It was initially developed by Web Consortium (W3C) and Internet Engineering Task Force (IETF). It allows to transfer videos and audios between different browsers. This research paper, analyse the parameters during the call in different browsers and conditions (number of end points). The concept of WebRTC is inspired from Session Initiation Protocol (SIP). It helps in the establishment of sessions and maintain it. It also supports data and message transmissions. It also works on remote location and different network transmission protocols. It also allows peer to peer communication. In this research work, we examine the behaviour of WebRTC and SIP during the call from different browsers. We examine the different parameters like packets sent, jitter, VO-Width and bandwidth during the call and call supported on cloud during our experimental work.

Index Terms: WebRTC, SIP, SDP, UDP, Codec, VoIP, Session Management, Internet Engineering Task Force (IETF), TLS, Channel Bitrate, Inter-Process Communication (IPC).

I. INTRODUCTION

WebRTC is a project that was started by google. It is a collection of frame-work and libraries. It is an open source and provides real-time communication between various web browsers and mobile applications. It uses simple application programming interfaces (APIs). It permits audio and video communication (VoIP). It does not use any third party software or plug-in. The session Initiation Protocol (SIP) is a signalling protocol. It is used for establishment, invite, maintaining, and terminating of media sessions between user agents or the end points. SIP is an application layer protocol used for delivery of voice and multimedia of internet telephony for voice and video calls, over the networks.

The main aim of this research work is to analyse the performance of these two frameworks in various scenarios. The flow of the paper is organized as follows: Section 2 shows the brief introduction of the proposed protocols, Section 3 describes Related Work, and Section 4 describes motivation of research work. In section 5, Experimental Setup is shown with the help of different scenarios. Results are discussed in

section 6, which is followed by Conclusion and Future Work in Section 7.

II. INTRODUCTION OF PROTOCOLS

Previously, we needed external support for implementation of media transfer using IP. Protocols like WebRTC and SIP make our task easier by using multiple standards, protocols and rich APIs. We firstly discuss the architecture of these protocols i.e. SIP and WebRTC which proceeds as below.

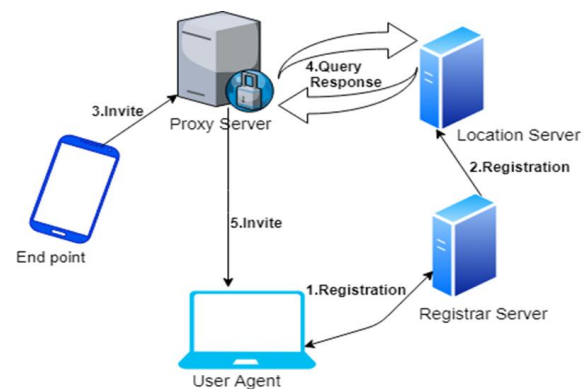


Fig.1 SIP Network Components

A. Session Initiation Protocol

Session Initiation Protocol (SIP) is a signaling protocol for video conferencing, VoIP, multiplayer games and real time messaging application. It lays set of rules how two systems communicate with one another by initiating sessions. It is rich repository of methods defined for different locations end points using different media interactions and capabilities. SIP is loose coupled and interactive protocol. It is developed and being looked upon by an IETF, which is society for developing Internet standards. It works on application layer and is also independent of network layers protocol. SIP network architecture can be understood as in Fig. 1. End-user may be client or server. It interacts with the proxy server, which functions similar to the router and forwards request to registrar server. Proxy server can be stateful or stateless having information of the network and other without any network information. Registrar server authenticates the end user and responds 200 Ok for success after which request goes to Location server as in Fig. 2.

Revised Manuscript Received on June 15, 2019.

Navrattan Parmar, parmarnavrattan@gmail.com, Department of Computer Engineering, National Institute of Technology, Kurukshetra, Haryana.

Virender Ranga, virender.ranga@nitkkr.ac.in, Department of Computer Engineering, National Institute of Technology, Kurukshetra, Haryana.

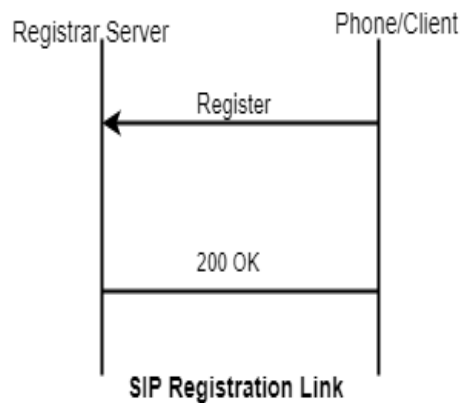


Fig. 1 Registration in SIP

location server replies 3xx (redirect response) from the location database created by the registrar server and session is established for communication. this whole process is depicted in Fig. 3.

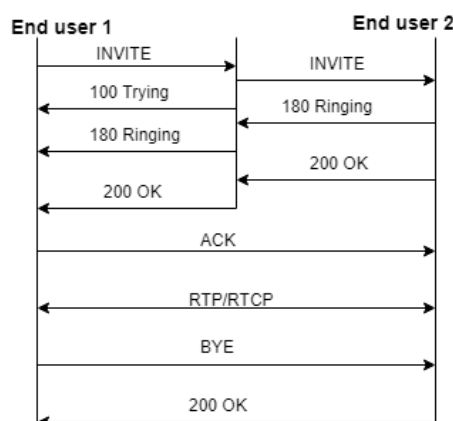


Fig. 2 SIP Session Establishment

SIP performs various function on session like Call drop, Hold, Video Conferencing, moderator meetings (controlled by leader), moderator-less. SIP allows Peer-to-Peer(P2P) as well Client-Server communication in a network. In Client-Server communication, unlike the P2P end-users any can have different capabilities. In P2P end users can swap their roles and if one is unavailable other can communicate. However, this allows different users to connect and disconnect during the call multiple times. SIP have different response codes which are important to remember for communication which can categorized as follows:

- 1** is class of response for End to End responses codes like 100 TRYING.
- 2** is class of response are for the response accepted.
- 3** It is a redirect response class by redirect server in reply to INVITE.
- 4** class of response are for Client failure similar to HTTP.
- 5** It is the response used for Server Failure.
- 6** It is class for Global failure Response. Server denies such request to be forwarded as it will fail at other location too.

SIP calling using SSIP is illustrated in Fig. 4, it can be seen SIP number are similar to DNS as explained.

B. WebRTC

WebRTC is an open source technology that allows peer-to-peer connection to different web applications. It is used for Real Time Communication (RTC) for conferencing, social networking, online medical consultations, live games etc. It provides RTCDataChannel, RTCPeerConnection, and MediaStream objects for the development. WebRTC architecture is defined as in Fig. 5.

- RTCPeerConnection: It provides connection to peers and attach audio/video's streams(onAddStream event). It use Java-Script Session Establishment Protocol (JSEP) for connection establishment. Secure Real-Time Transport Protocol(SRTP) is used for the making the media delivery secure and safe. It also provides Secure Real-Time Control Transport Protocol(SRTCP) for reliability[22].
- MediaStream(getUserMedia api): It request for the media stream and adds it to RTCPeerConnection. it establishes the connection using the signaling server firstly and there after configures for direct communication.
- RTCDataChannel: It is used to send data along with the media streams. It uses Stream Control

FIG. 4 SIP CALL

```

No call Running
INFO: No Call Running, Proceed with Call Dial
meetingId:arysks
INFO: 731689417.6754
INFO: Meeting IDs [731689417.6754]
INFO: Con node to dial arr 219.65.87.38
INFO: Con Node to Dial 219.65.87.38
INFO: Moderator Or Participantparticipant
INFO: SIP Or h323Sip
INFO: Direct Dial or IVR Dial: directDial
Tandberg API: directDial
Tandberg API: Sip
Tandberg API: 731689417
Tandberg API:
Tandberg API: 219.65.87.38
INFO: xcommand dial Protocol: Sip Number: 731689417@219.65.87.38 CallRate: 6000
xcommand dial Protocol: Sip Number: 731689417@219.65.87.38 CallRate: 6000
Execute With Agra: xcommand dial Protocol: Sip Number: 731689417@219.65.87.38 CallRate: 6000
Execute With Agra: admin
Execute With Agra: BlueJe@n$
Results: spawn /bin/bash
bash-3.2$ ssh admin@219.65.87.37
Password:
Welcome to tandberg
Cisco Codec Release TC7.3.17.52b35e9
SW Release Date: 2019-03-18
#r Login successful
OK
xcommand dial Protocol: Sip Number: 731689417@219.65.87.38 CallRate: 6000
    
```

Transmission Protocol (SCTP) for secured data transfer. It adds TCP like features like multiplexing, flow control and reliability[23].

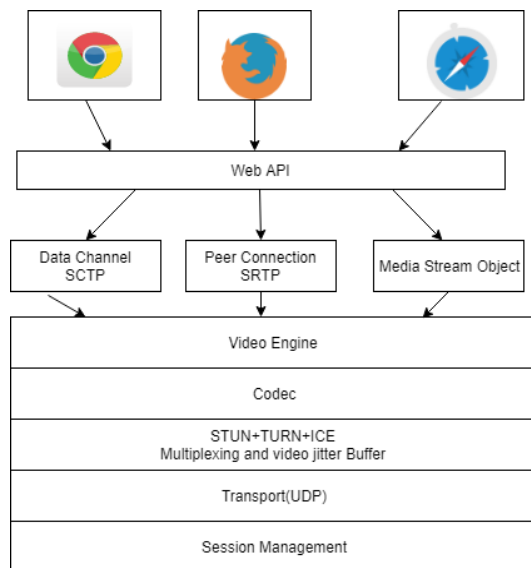


Fig. 3 WebRTC Architecture

From the End User Media Stream is split into codecs like H.264, ISAC, OPUS and VP8. Different browsers may use different codecs for audio and video. WebRTC transport use UDP because loss of packets is not that significant. We can still maintain a video quality even after the packets loss to a certain value. It also establishes and maintains the sessions.

III. BACKGROUND STUDY

Tomokatsu Mizukusa et. al [2] propose an environment for SIP products based on Feed-Forward design. Its environment gives performance value that includes Signal Processing and Thermal Dispersion. This research results in reduction of design period.

Dirk Hoffstadt et. al[3] gave the architecture, features and usage of a sip trace recorder(str) used to parse and store important sip data in database. str plug-ins provides threat analyses also with privacy option.

Helmut Hlavacs et. al[4] describe an approach Babel-SIP for increasing the rate of acceptance for SIP messages and concludes that, Babel-SIP can exceedingly enhance the message acceptance Rate. The main motto of Babel-SIP is to act as intermediary for a SIP proxy and analyzes its messages sent to its proxy. It studies which register messages were accepted by this proxy, and the ones rejected.

Demir Y. Yavas et. al[5] present a Fluid-Flow model to analyze the Priority-Based Request Scheduling Mechanism (PRSM) in the overloaded SIP server having an infinite buffer. The model justify the performance of the PRSM Fluid-Flow Model.

Cağatay Vildiz et. al[6] show the Real Time Sip Network Simulation and Monitoring System. In Simulation System is installed at a SIP server, contribute services for assembling network data and server statistics. It also gives a framework for developing SIP Network Applications.

Jin Zhou et. al[7] propose an approach of automated SIP network discovery based on message probing. This approach gives minimal number of SIP request messages that will be sent out to probe SIP entities. It could then wrap and parse the received responses and required information could be obtained. while costs have been limited probing messages, their results show that the approach can effectively obtain sip information for management.

Gao Zhiguo et. al[8] presents SIP accelerator (SIP Offload Engine (SOE)) to enhance server performance. It helps offloading parsing, processing, security and transport. Research results shows improvement in the SIP Server throughput.

Abhishek Bansal et. al[9] focuses on DOS attack by SIP messages and analyze server performance . Calculates CPU Utilization and memory usage during VoIP calls by using the performance metric. Results shows when SIP server is overloaded by call requests so that quality of call degraded.

Victoria Beltran et.al[10] focuses on different IDM Models. This research is on cloud-based services. It manages user utilities in Unified Communication as Service(UCaaS). This research evaluates requirements on WebRTC-based UC services and propose modifications of WebRTC to meet the requirements on IDM. Analysis of different models to identify information storage. It accelerates the information recast from the enterprise to the cloud.

Alfonso Sandoval Rosas et. al[11] define the meeting(video session), i.e. system oriented and gives users higher understandability to communicate by convey standard telephony and multimedia in real-time on web browsers. This research proposes a cooperative interaction scheme that differs from others in the context of telephone network communication straight from a web browser during an active video conference, which also enables Real-Time Media Streams between these two techniques to be exchanged.

Basar Daldal et. al[12] propose an approach to make communication between WebRTC-to-WebRTC, and WebRTC-to-legacy VoIP by

the use of Restful Web Services. Its approach is to carry "a pointer" that contains a value to define a Restful Service link to actual media description of endpoints instead of media description itself.

Ahmadreza Montazerolghaem et. al[13] paper implements OpenSIP on a real testbed which includes open VSwitch and the floodlight controller. The results show that the proposed architecture has a low overhead and satisfactory performance and also take benefit of a flexible scale-out design during application deployment.

katrien De Moor et. al[14] conducts an experiment with 22 observations where 2-party WebRTC based audiovisual conversations took place under varying technical conditions. It is observed that the data and unity of quality of experience of WebRTC is affected by the annoyance leads to lowest quality.

Huaying Xue et. al[15] paper propose a framework based on WebRTC, video conferencing system with augment features of screen sharing. It explains architecture with its components. This approach gives a premium Quality video even in the low bandwidth networks and ensures wonderful user experience.

Alexandre Gouaillard et. al[16] provide a thorough overview of the various Testing Problems encountered when the WebRTC was first released. Testing is carried out on the grounds of compliance with W3C. It tests Node.js API, Stand-Alone Web Application Testing, WebRTC Safety testing, P2P Network and ICE testing, Synchronous and Asymmetric Testing and complete worldwide interoperability Testing.

IV. MOTIVATION

- Md.I.Hussain et. al[1] propose a detailed study of various issues in IoT like scalability, interoperability, heterogeneity, quality of service and security. the proposed research work focuses on interoperability, and QOS.

- It will be useful for Testing and Automation for SIP and WebRTC protocols on cloud.
- This research provides a lot of useful observation's on QOS parameters.
- It will be boon to the student's and researchers as it will give them a direction and quick start in world of IoT. They will understand the architecture and QOS parameters for SIP and WebRTC.
- This work will not only help in understanding, but also make the us aware of broad scope of WebRTC.
- WebRTC and SIP are having high cohesion and low coupling. It makes the development easy. There are tremendous number of classes available we just need to implement them. These protocols are Open source and also reliable, so development is trusted.
- It is possible for an individual to design a video calling application on top of WebRTC.

V. EXPERIMENTAL SETUP

Here we analyze, the call through WebRTC[17][18] and SIP. Our proposed approach uses different browsers, different number of end points at different locations as in shown in Fig.6, Call through WebRTC. It has been deduced that the capacity of the channel is directly proportional to the power of the signal when the bandwidth remains constant. Chrome browser use codec vp-8 for video and opus codec for audio. Similarly different browsers use different codec for call quality[19][20]. It may be noted different versions of same browsers can also use different codecs.

A. Scenario 1

We join the call using two end points at remote location via SIP[21] protocol. This experiment use chrome and Firefox web-browser. In this Scenario, Call did not connect only one end point was there. We analyse the value of different parameter and found bitrate is quite low, close to zero. Audio Bitrate and Video Bitrate decreases to zero(since,failure).

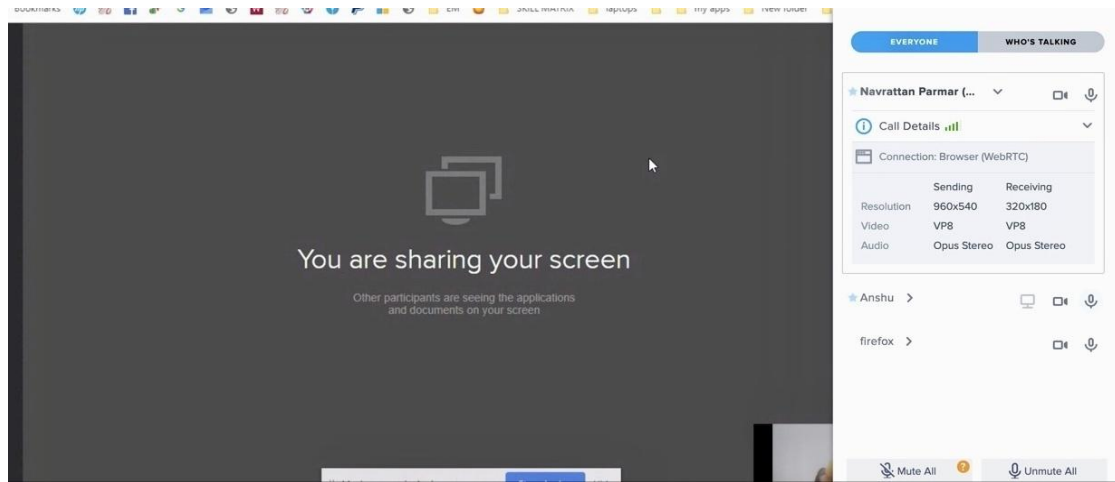


Fig. 4 Call through WebRTC

B. Scenario 2

We join the call using two end points at local location via SIP protocol. We use chrome and firefox web-browser. In this scenario call connects . We can show it in Fig.7, Fig.8, Fig. 9, and Fig. 10.

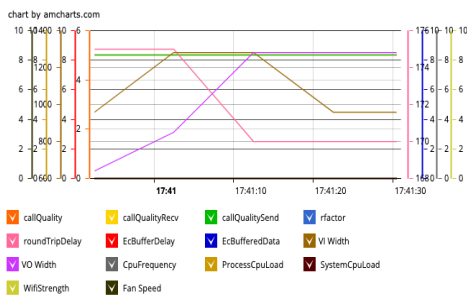


Fig. 7. End-Point in Scenario-2

- I. Audio Bitrate and Video Bitrate remains constant.
- II. VI Width =1280 and VO Width=1200 means quality is 720p.
- III. RTT is 275 for video and 247 for audio. No Network Loss.
- IV. No Network loss.
- V. Call Quality is constant.
- VI. Channel Rate is 92000.

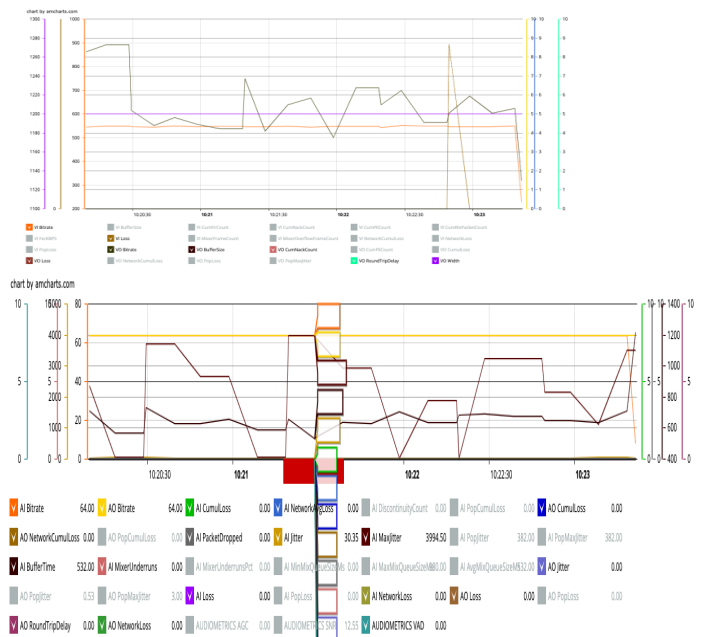


Fig.8 Audio in Scenario-2 &

Fig.9 Video in Scenario-2

```

xstatus Diagnostics Call Channels IncomingVideoChannel
*# Diagnostics Call 70 Channels IncomingVideoChannel 1078 Netstat 1 Bytes: 5288167
*# Diagnostics Call 70 Channels IncomingVideoChannel 1078 Netstat 1 ChannelRate: 92000
*# Diagnostics Call 70 Channels IncomingVideoChannel 1078 Netstat 1 Drop: 0
*# Diagnostics Call 70 Channels IncomingVideoChannel 1078 Netstat 1 Jitter: 4
*# Diagnostics Call 70 Channels IncomingVideoChannel 1078 Netstat 1 LastIntervalLost: 0
*# Diagnostics Call 70 Channels IncomingVideoChannel 1078 Netstat 1 LastIntervalReceived: 90
*# Diagnostics Call 70 Channels IncomingVideoChannel 1078 Netstat 1 Loss: 1
*# Diagnostics Call 70 Channels IncomingVideoChannel 1078 Netstat 1 MaxJitter: 5
*# Diagnostics Call 70 Channels IncomingVideoChannel 1078 Netstat 1 Packets: 6799
*# Diagnostics Call 70 Channels IncomingVideoChannel 1080 Netstat 1 Bytes: 0
*# Diagnostics Call 70 Channels IncomingVideoChannel 1080 Netstat 1 ChannelRate: 0
*# Diagnostics Call 70 Channels IncomingVideoChannel 1080 Netstat 1 Drop: 0
*# Diagnostics Call 70 Channels IncomingVideoChannel 1080 Netstat 1 Jitter: 0
*# Diagnostics Call 70 Channels IncomingVideoChannel 1080 Netstat 1 LastIntervalLost: 0
*# Diagnostics Call 70 Channels IncomingVideoChannel 1080 Netstat 1 LastIntervalReceived: 0
*# Diagnostics Call 70 Channels IncomingVideoChannel 1080 Netstat 1 Loss: 0
*# Diagnostics Call 70 Channels IncomingVideoChannel 1080 Netstat 1 MaxJitter: 0
*# Diagnostics Call 70 Channels IncomingVideoChannel 1080 Netstat 1 Packets: 0
** end

```

Fig.10 Channel Rate in Scenario-2

Performance Analysis of WebRTC and SIP for Video Conferencing

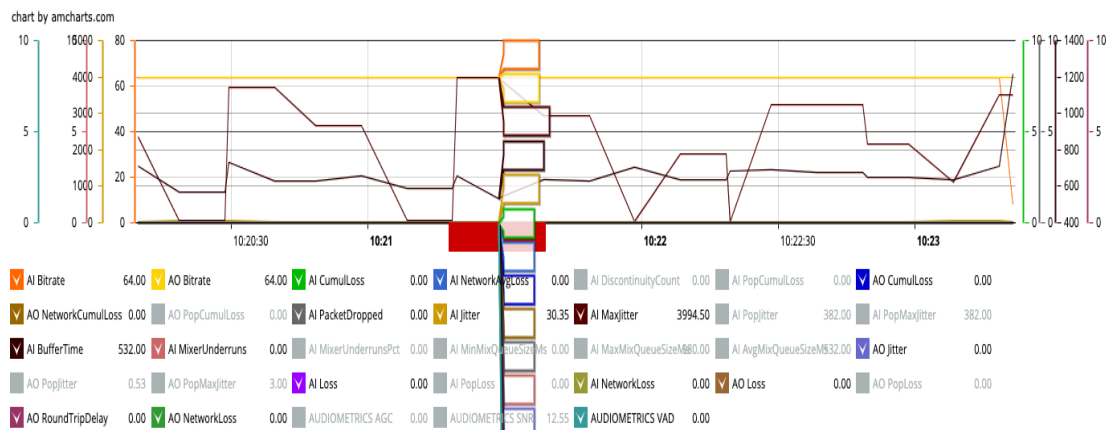


Fig.11 Audio in Scenario-3

C. Scenario-3

We join the call using 22 end points at local location via SIP protocol. We use chrome and Firefox web-browser. In this scenario call connects to all . We observe as in Fig.11, Fig.12, Fig.13.

- I. Audio Bitrate and Video Bitrate remains constant.
- II. VI Width =1920 and VO Width=1280 means quality is 1080p(HD).
- III. RTT is 269 for video and 244 for audio(average).
- IV. NO Network Loss.
- V. Call Quality is constant.

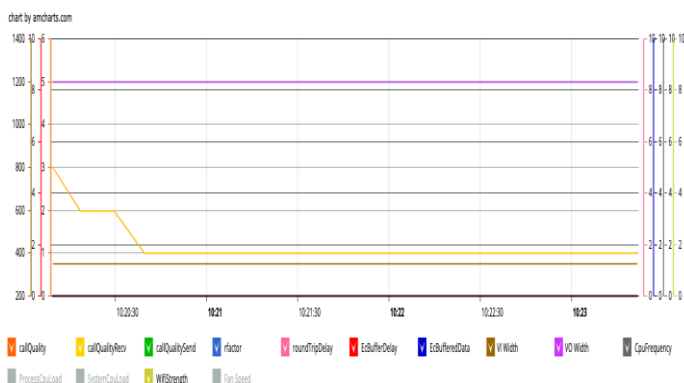


Fig.12 An End-point in Scenario-3

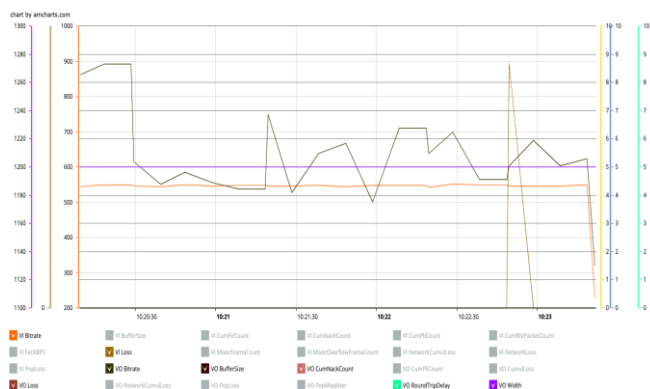


Fig.13 Video in Scenario-3

D. Scenario- 4

We join the call using two end points at local location via WebRTC[22][23] protocol. We use chrome and firefox web-browser. In this scenario call connects to both end-points .We observe the results as in Fig.14.

- I. Audio Bit-rate and Video Bit-rate remains constant
- II. VI Width=1920 and VO Width=1920 means quality is 1080p(HD) for both send and receive.
- III. RTT is 267 (average).
- IV. NO Network Loss.
- V. Call Quality is constant and is equal to call quality received.

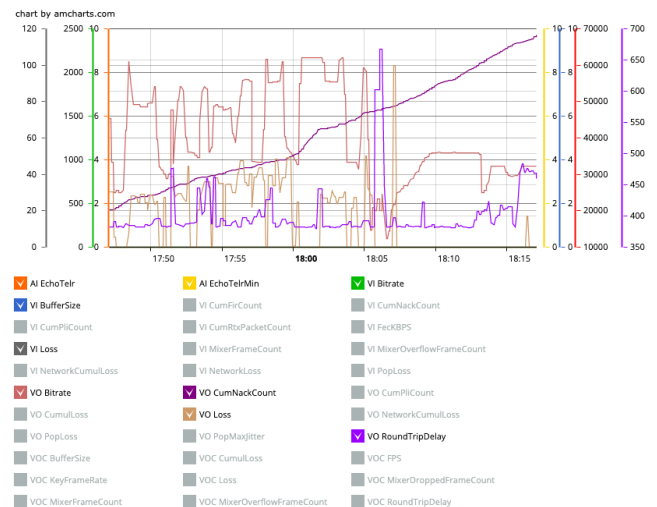


Fig.14 Call Quality in WebRTC in Scenario-4

E. Scenario-5

We join the call using two end points at remote location via SIP protocol and WebRTC protocol. We use chrome and Firefox web-browser. In this scenario call connects. At the starting of the call, video quality is low and audio remained almost constant during the call[24][25]. We observe following observations in Fig.15, Fig.16 and Fig.17.

- I. Audio Bitrate and Video Bitrate remains constant
- II. VI Width = 1200 and VO Width = 1200 means quality is 720p. for both send and receive.
- III. At SIP, AO Round Trip Delay is

33.94ms, VO Round Trip Delay is 50.61 and round trip delay average is 46.83.

- IV. At webRTC, AO Round Trip Delay is 30.67ms, VO Round Trip Delay is 20.75 and round trip delay average is 28.74.
- V. No Network Loss.
- VI. Call Quality is constant and is equal to call quality received.

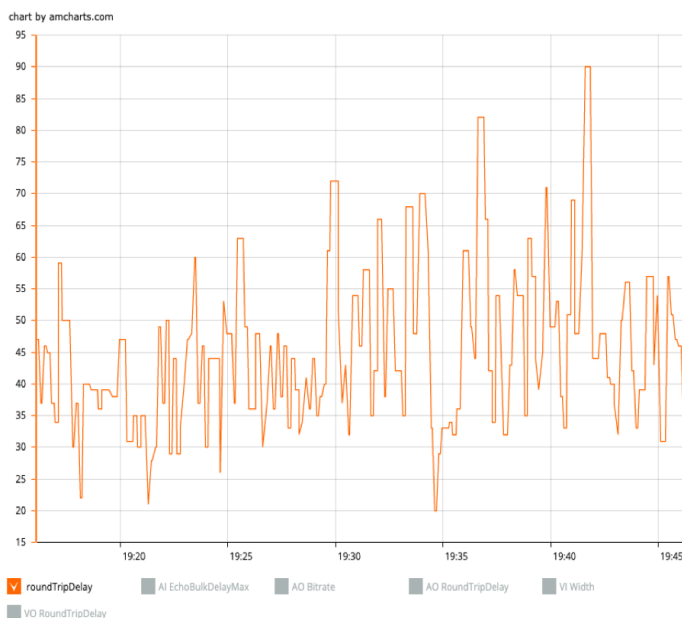


Fig.15 SIP end Point in Scenario-5

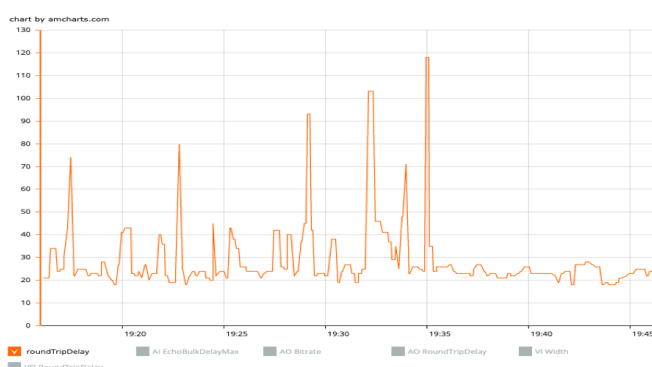


Fig. 16 WebRTC end point in Scenario-5 diag-1



Fig.17 WebRTC end Point in Scenario-5 diag-2

VI. RESULT DISCUSSION

After the repeated observations, some conclusions may be drawn:

- There is no significant network loss (AI,AO,VI,VO or any other) or error loss. So, it not a major problem be looked in.
- Even though some packets are dropped, but the call quality remains almost constant. We can use UDP as transport protocol in Real Time Communication(RTC's).
- Call quality is better in WebRTC protocol on the same network conditions. RTT is independent on Bitrate.
- There is little or no difference is calling from remote location than local locations.
- Bandwidth effects the call quality.
- Channel rate is directly proportional to SNR, thus channel rate is dependent on Power of Signal and Bandwidth.
- Latency is low which is the advantage and why we use these protocols for RTC.

VII. CONCLUSION

- I. There are several other protocols like DDS which can built on top of UDP and can provide RTC. In the future, we will understand its architecture and implement for different scenarios.
- II. WebRTC is being developed by open source community and it can be extended support to android and even mobile browsers.
- III. As we observed, there is a difference in RTT in audio stream and video stream it can be combined in a single stream.

REFERENCES

1. Hussain, Md.I, Internet of Things: Challenges and Research Opportunities, *published in International conference on dependable systems and networks workshops* 35, 123-126.
2. Mizukusa ,T., Nagano, T. , Shimizu, Y, Sakata, K. & Kato, K. (2009), Development of Feed-Forward Design System for Rapid SiP Design, *IEEE International Conference on 3D System Integration, San Francisco, CA*, 1-4.
3. Hoffstadt, D. Monhof, S. & Rathgeb, E. (2012), SIP Trace Recorder: Monitor and Analysis Tool for threats in SIP-based networks, *8th International Wireless Communications and Mobile Computing Conference (IWCMC), Limassol*, 631-635.
4. Hlavacs, H., Hummel, K.A., Hess, A. & Nussbaumer, M. (2008), Babel-SiP: Self-learning SIP Message Adaptation for Increasing SIP-Compatibility, *IEEE INFOCOM Workshops 2008, Phoenix, AZ*, 1-6.
5. Vavas, D.V., Hokelek, I. & Günsel, B. (2018), On modeling

of priority-based SIP request scheduling, *Simulation Modelling Practice and Theory*, vol 80, 128–144.

6. Yildiz C., Kurt,A.B, Ceritli,T.Y., Sankur,B & Cemgil,A.T.(2018), A real-time SIP network simulation and monitoring system", *SoftwareX*, vol.8 ,21–25.

7. Zhou,J., Li,J., Xia,Y.B., Cai,B., & Ying,C.(2008), SIP Network Discovery by Using SIP Message Probing, *IEEE Network Operations and Management Symposium, Salvador, Bahia*, 791-794.

8. Zhiguo,G., Zhe,X.,Wei,X., Zhiyong,L., & Bo,Y.(2009), SIP Offload Engine for Accelerating J2EE Based SIP Application Server, *International Conference on Communication Software and Networks, Macau*, 749-753.

9. Bansal,A., Kulkarni,P., & Ais,A.R.(2013), Effectiveness of SIP Messages on SIP Server, *IEEE Conference on Information & Communication Technologies, Thuckalay, Tamil Nadu, India*, 616-621.

10. Beltran,V. & Bertin,E (2015), Unified communications as a service and WebRTC: An identity-centric perspective, *Computer Communications*, 73–82.

11. Rosas,A.S. & Martínez,J.L.A.(2016), " Videoconference System Based on WebRTC With Access to the PSTN, *Electronic Notes in Theoretical Computer Science*, vol 329,105–121.

12. Daldal,B., Bilgin,I., Basaran,D. & Metin,S.(2016), Using Web Services For WebRTC Signaling Interoperability, *IEEE/IFIP Network Operations and Management Symposium, Istanbul*, 780-783.

13. Montazerolghaem, A., Moghaddam,M.H.Y., & Leon-Garcia,A.(2018), OpenSIP: Toward Software-Defined SIP Networking , in *IEEE Transactions on Network and Service Management*, vol. 15,no.1, 184-199.

14. Moor,K.D., Arndt,S. & Ammar,D.(2017), Exploring diverse measures for evaluating QoE in the context of WebRTC, *Ninth International Conference on Quality of Multimedia Experience (QoMEX), Erfurt*, 1-3.

15. Xue,H. & Zhang,Y. (2016) A WebRTC-Based Video Conferencing System with Screen Sharing, *2nd IEEE International Conference on Computer and Communications (ICCC), Chengdu*, 485-489.

16. Gouaillard,A. & Roux,L.(2017), Real-Time Communication Testing Evolution with WebRTC 1.0, *Principles, Systems and Applications of IP Telecommunications (IPTComm), Chicago, IL*, 1-8.

17. Jian.C. & Lin,Z.(2015), Research and Implementation of WebRTC Signaling via WebSocket-based for Real-time Multimedia Communications, *5th International Conference on Computer Sciences and Automation Engineering ICCSAE*, 374-380.

18. Edan,N.M., Al-Sherbaz,A. & Turner,S.(2017), Design and Evaluation of Browser-to-Browser Video Conferencing in WebRTC", *Global Information Infrastructure and Networking Symposium (GIIS), St. Pierre*, 75-78.

19. Haensge,K., & Maruschke,M.(2015), QoS-based WebRTC Access to an EPS Network Infrastructure", *18th International Conference on Intelligence in Next Generation Networks, Paris*, 9-15.

20. Kim,W., Jang,H., Choi,G., Hwang,I. & C. Youn,(2016), A WebRTC based live streaming service platform with dynamic resource provisioning in cloud, *IEEE Region 10 Conference (TENCON), Singapore*, 2424-2427.

21.. Zubair,M., Kong,X., Jamshed,I., & Ali,M.(2014), Integrating SIP with F-HMIPv6 to Enhance End-to-End QoS in Next Generation Networks, *Advances in Intelligent Systems and Computing* 240, Springer International Publishing Switzerland.

22. Yan,S, Guo,Y. Y.Chen, & Xie,F.(2019), Predicting Freezing of WebRTC Videos in WiFi Networks, *ICST Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, Published by Springer Nature Switzerland AG 2019*, 292–30.

23. Rodríguez,P., Cerviño,J., Trajkovska,I. & Salvachúa,J.(2019),"Advanced VideoConferencing Services Based on WebRTC", *Conference: IADIS Multi Conference on Computer Science and Information Systems*.

24. Nayyef, Z.T., Amer, S.F. & Hussain,Z.(2018),Peer to Peer Multimedia Real-Time Communication System based on WebRTC Technology, *International Journal of Engineering & Technology*, 125-130.

25. Zafran, M.R.M., Gunathunga, L.G.K.M. , Rangadhari, M.I.T, Gunarathne, M.D.D.J, Kuragala K.R.S.C.B, Dhammearatchi, M.D.(2016), Real Time Information and Communication Center based on webRTC, *International Journal of Scientific and Research Publications, Volume 6, Issue 4*, 644-649.

AUTHORS PROFILE



Navrattan Pamar is currently pursuing M.Tech, Computer Engineering from National Institute of Technology, Kurukshetra, Haryana, India. He has graduated from Guru Nanak Dev University, Amritsar in 2017. He has been published one book chapter in Springer Lecture Notes in Networks and Systems, (Scopus Indexed). He is currently looking forward in research in protocols and currently working as an Intern in a software company in Bangalore.



Virender Ranga received his PhD degree in 2016 from Computer Engineering Department of National Institute of Technology, Kurukshetra, Haryana, India. He has published more than 50 research papers in various International SCI Journals and reputed International Conferences in the area of Computer Communications. Presently, he is Assistant Professor in the Computer Engineering Department since 2008. He has been conferred by Young Faculty Award in 2016 for his excellent contributions in the field of Computer Communications. He has been acted as member of TPC in various International conferences of repute. He is a member of editorial board various reputed journals like Journal of Applied Computer Science & Artificial Intelligence, International Journal of Advances in Computer Science and Information Technology(IJACSIT), Circulation in Computer Science (CCS), International Journal of Bio Based and Modern Engineering (IJBME) and International Journal of Wireless Networks and Broadband Technologies. He is an active reviewer of many reputed journals of IEEE, Springer, Elsevier, Taylor & Francis, Wiley and InderScience. His research area includes Wireless Sensor & Ad-hoc Networks, SDN, IoT security, FANET security.

