International Journal of Innovation in Computer Science and Information Technology
Vol.4, NO.1 , P:40 - 50
Received: 21 March 2022
Accepted: 10 June 2022

INTERNATIONAL ORGANIZATION OF IOAS ACADEMIC STUDIES

A research on multimedia information security in WebRTC

Hossein Moradi
Technical and Engineering – University of Science and Culture

Abstract

The purpose of this research paper is to provide an in-depth analysis of multimedia information security in WebRTC (Web Real-Time Communication). WebRTC has attracted considerable attention for its ability to enable real-time communication between web browsers, leading to various multimedia applications and services.

However, ensuring the security and privacy of multimedia information transmitted through WebRTC poses significant challenges.

This paper examines the strengths of WebRTC, discusses the challenges facing multimedia information security, and provides approaches and solutions to mitigate these challenges. This research is based on an extensive review of the available literature and incorporates insights from industry experts and researchers. A total of 20 authoritative references have been used to support the discussion.

Key words: multimedia information, real-time web communication, information security, security

1. Introduction

Multimedia information security in WebRTC is a set of standardized technologies that enable real-time communication between web browsers without the need for plugins or installing additional software.

It provides an Integrated framework for the transmission of voice, video and data, revolutionizing various multimedia applications and services over the Internet. WebRTC incorporates open standards such as HTML5, JavaScript, and the Real-Time Protocol (RTP) to facilitate direct peer-to-peer communication between browsers, eliminating the need for intermediaries.

WebRTC offers several key features that contribute to its widespread adoption. First, it enables real-time, two-way communication, making it ideal for applications such as video conferencing, online gaming, and live streaming.

Second, WebRTC supports cross-platform compatibility and enables communication between different browsers and operating systems. This interoperability has significantly expanded the reach and accessibility of real-time multimedia communications.

While WebRTC brings tremendous benefits in real-time communication, ensuring the security and privacy of multimedia information transmitted through this technology is of utmost importance.

Multimedia data, including audio and video streams, can contain sensitive and confidential content, making them prime targets for unauthorized access, interception, or manipulation.

Protecting the integrity, confidentiality, and authenticity of multimedia information is critical to maintaining user trust and ensuring the successful adoption of WebRTC-based applications and services [1].

The purpose of this research paper is to investigate the field of multimedia information security in the context of WebRTC. It aims to examine the strengths, challenges and approaches associated with securing multimedia data sent via WebRTC.

2. Strengths of WebRTC

2.1 Real-time communication capabilities

WebRTC, short for Web Real-Time Communication, has a wide range of strengths that make it a powerful platform for real-time communication on the web. One of its key strengths lies in its ability to provide seamless real-time communication capabilities that allow users to engage in voice, video, and data exchanges in a coordinated manner. WebRTC eliminates the need to install additional plugins or software and enables direct peer-to-peer communication through supported web browsers [2].

2.2 Peer-to-peer architecture

Another strength of WebRTC is its peer-to-peer architecture, which facilitates direct communication between web browsers without the need for an intermediary or central server. This decentralized approach increases privacy and reduces latency by establishing direct communication between participants. Peer-to-peer communication ensures that multimedia information, such as audio and video streams, can be transferred directly between users, thereby minimizing bottlenecks and improving overall efficiency [3].

2.3 Cross-Platform Compatibility

WebRTC provides cross-platform compatibility and enables real-time communication between different operating systems, web browsers, and devices. This compatibility extends to desktops, laptops, tablets and mobile devices, ensuring that users can engage seamlessly regardless of their platform of choice. Whether it is a Windows PC, macOS, Linux, Android, or iOS device, WebRTC provides a unified framework for multimedia communication and increases accessibility and convenience for users [4].

2.4 Scalability and flexibility

Scalability and flexibility are essential strengths of WebRTC. WebRTC supports dynamic scalability, allowing applications to adapt to varying network conditions and efficiently handle large numbers of concurrent users. This scalability is especially important for multimedia applications, such as video conferencing, live streaming, and online gaming, where the number of participants can vary significantly. In addition, WebRTC provides flexibility in terms of deployment options, enabling integration with existing communication infrastructure or the development of stand-alone applications [5].

As a result, WebRTC has several strengths that make it a powerful platform for real-time communications on the Web. Its real-time communication capabilities enable synchronized audio, video and data exchange, while the peer-to-peer architecture ensures direct and efficient communication between browsers.

Cross-platform compatibility allows for seamless communication between different operating systems and devices, increasing accessibility. Finally, the scalability and flexibility of WebRTC allows applications to handle different network conditions and adapt to different scenarios. These strengths collectively contribute to the popularity of WebRTC and its wide range of multimedia applications in today's digital landscape [6].

3. Challenges in multimedia information security

3.1 Encryption and Authentication

In the area of multimedia information security in WebRTC, encryption and authentication are very important aspects to consider. Encryption ensures that transmitted multimedia data remains confidential and cannot be intercepted by unauthorized entities. On the other hand, authentication verifies the identity of the participants involved in the communication and prevents impersonation and unauthorized access. However, there are several challenges in implementing strong encryption and authentication mechanisms in WebRTC [7].

One of the challenges lies in choosing the right encryption algorithms and key management protocols. WebRTC supports various encryption algorithms, including

Received: 21 March 2022
Accepted: 10 June 2022

AES (Advanced Encryption Standard) and DTLS (Data Transport Layer Security). However, the selection of algorithms should consider factors such as computational efficiency, security level, and compatibility with different devices and browsers. Key management protocols, such as SRTP (Secure Real-Time Transport Protocol) and SDES (Secure Description) are also essential for secure multimedia transmission. Ensuring secure exchange and management of encryption keys is very important to prevent unauthorized access to multimedia content [8].

3.2Privacy Concerns

Privacy concerns are very important in multimedia information security, especially given the sensitive nature of the transmitted data. WebRTC applications often involve audio and video communications, raising privacy concerns related to content visibility and user identification. The challenge lies in preserving users' privacy while enabling seamless communication [9].

Various techniques can be used to address privacy concerns. Anonymization techniques, such as masking the user's IP address and using aliases, help protect the user's identity. Additionally, selective encryption allows certain parts of a multimedia stream to be encrypted while leaving other parts visible, ensuring privacy without compromising application performance. However, striking the right balance between privacy and usability remains a challenge, as too much privacy can hinder communication effectiveness.

3.3Network Vulnerabilities

WebRTC operates in a distributed network environment, making it susceptible to various network vulnerabilities. Attacks such as man-in-the-middle attacks, eavesdropping, and packet eavesdropping are important threats to multimedia information security. WebRTC relies on various protocols to establish and maintain communication channels, including ICE (Interactive Connection Establishment) and STUN (Setup Traversal Utilities for NAT). However, these protocols may have vulnerabilities that can be exploited by attackers. To reduce network vulnerabilities, it is necessary to implement secure network protocols such as DTLS and SRTP. Secure configuration of network infrastructure, including firewalls and routers, also helps prevent unauthorized access to communication channels. Regular monitoring and intrusion detection systems can identify suspicious network activities and take appropriate actions to maintain the security of multimedia transmission [10].

3.4 Honesty and media manipulation

Ensuring the integrity of multimedia content is another critical challenge in WebRTC. Attackers may attempt to manipulate the transmitted multimedia data resulting in content modification, malicious code injection, or unauthorized modifications. Maintaining the integrity of the multimedia stream is very important to maintain the integrity and reliability of the exchanged information. To address this challenge, cryptographic hash functions can be used to verify the integrity of multimedia content. By calculating the hash values of the data sent at the sender's end and comparing them with the data received at the receiver's end, any tampering or alteration can be detected. In addition, digital signatures can be used to ensure the authenticity and integrity of multimedia content and provide a means of verifying the source of data.

3.5 Denial of Service Attacks

Denial-of-Service (DoS) attacks are a significant challenge for multimedia information security in WebRTC. The aim of these attacks is to disrupt the availability and quality of multimedia communications and make the service unusable. DoS attacks can be implemented in a variety of ways, including flooding the network with excessive traffic, exploiting vulnerabilities in the WebRTC stack, or compromising server infrastructure.

To mitigate the impact of DoS attacks, WebRTC applications must implement robust measures to detect and mitigate such attacks. Traffic monitoring and anomaly detection techniques can identify abnormal traffic patterns and initiate appropriate countermeasures. Load balancing and scaling mechanisms can help distribute incoming traffic and prevent system frustration. In addition, implementing rate limiting and session control mechanisms can limit the number of concurrent connections and prevent resource exhaustion.

Consequently, securing multimedia information in WebRTC involves addressing several challenges. Encryption and authentication, privacy concerns, network vulnerabilities, media integrity, and denial-of-service attacks all require attention.

By implementing appropriate security measures, such as encryption algorithms, privacy enhancement techniques, secure network protocols, integrity verification mechanisms, and robust DoS mitigation strategies, the security of multimedia information transmitted via WebRTC can be significantly enhanced.

4. Approaches to increase the security of multimedia information

4.1 Secure Key Exchange Mechanisms

Secure key exchange mechanisms play a vital role in ensuring the confidentiality and integrity of multimedia information in WebRTC. Establishing a secure communication channel between participants involves the exchange of encryption keys. However, this process must be protected from eavesdropping.

One of the common secure key exchange mechanisms is the Diffie-Hellman key exchange protocol. This allows two parties to securely exchange encryption keys over an insecure channel without the risk of key interception.

This protocol uses mathematical algorithms to generate a shared secret key that can be used for symmetric encryption, thus ensuring the confidentiality of multimedia data.

Other key exchange mechanisms include the Elliptic Curve Diffie-Hellman (ECDH) protocol, which increases security and efficiency, and the RSA (Rivest-Shamir-Adleman) algorithm, which uses public key cryptography. These mechanisms provide strong protection against key exchange vulnerabilities and ensure secure multimedia communications.

4.2 Encryption and cryptography protocols

Encryption is a fundamental approach to enhance the security of multimedia information in WebRTC. By encrypting the transmitted multimedia data, it becomes unintelligible to unauthorized entities, maintaining confidentiality and preventing data breaches.

WebRTC supports various encryption and decryption protocols such as AES (Advanced Encryption Standard) and DTLS (Data Transport Layer Security). AES is widely used for symmetric encryption and ensures that data remains confidential during transmission. DTLS, based on the Transport Layer Security (TLS) protocol, adds an additional layer of security to WebRTC by encrypting the media transfer.

In addition to encryption algorithms, cryptographic protocols such as SRTP (Secure Real-Time Transport Protocol) and SDES (Secure Description) provide mechanisms for securely transmitting and receiving multimedia data. These protocols provide end-to-end encryption that protects the integrity and confidentiality of multimedia information exchanged in WebRTC applications.

4.3 Secure Authentication Mechanisms

Secure authentication mechanisms are critical to ensure that only authorized participants can access multimedia communications in WebRTC.

Authentication verifies the identity of users, prevents impersonation and unauthorized access to the multimedia stream. One of the widely used authentication mechanisms is the use of digital certificates.

Digital certificates provide a means to verify the identity of participants using a trusted third-party certification authority.

Participants provide their digital certificates during the authentication process and the certificates are verified to ensure the authenticity of the users.

Another method is to use secure tokens or access tokens. These tokens are generated during the authentication process and are used to authenticate subsequent requests, ensuring that only authenticated users can access multimedia communications.

4.4 Media integrity protection techniques

Protecting the integrity of multimedia data is essential to ensure that it remains unchanged during transmission. The purpose of media integrity protection techniques is to identify any unauthorized changes or manipulations in multimedia content.

One common approach is to use cryptographic hash functions, such as SHA-256 (256-bit Secure Hash Algorithm).

Hash functions generate a fixed-size hash value unique to the input data. Any change or manipulation can be detected by calculating the hash value of the multimedia data sent and comparing it with the received data.

Digital signature is another technique to ensure media integrity. Digital signatures are generated using cryptographic algorithms and provide a means of verifying the authenticity and integrity of multimedia content. The recipient can verify the digital signature using the sender’s public key and ensure that the content has not been altered.

4.4 Privacy Enhancing Technologies

Privacy-enhancing technologies aim to address privacy concerns in multimedia information security. These technologies help protect the privacy of participants and ensure that sensitive information remains confidential during WebRTC communication.

It is a selective encoding approach where only certain parts of the multimedia content are encoded. By encrypting sensitive parts, such as the video stream or certain metadata, privacy can be maintained while leaving other parts visible for application purposes.

Anonymization techniques also help increase privacy. Techniques such as masking a user’s IP address, using pseudonyms, or using anonymous routing protocols help protect the identity of participants and make it difficult for unauthorized entities to track or identify them.

4.5 Network security measures

Network security measures are critical to protect against network-based attacks and vulnerabilities in WebRTC. These measures ensure the confidentiality, integrity and availability of multimedia communications.

Implementing secure network protocols, such as TLS/SSL (Transport Layer Security/Secure Sockets Layer), helps to establish secure communication channels between participants. Secure configuration of firewalls and routers, along with intrusion detection systems, provide additional layers of protection against unauthorized access and network attacks.

Load balancing and scalability mechanisms also contribute to network security. By distributing incoming traffic and preventing resource exhaustion, these measures help maintain the availability and performance of the multimedia communication system. As a result, increasing the security of multimedia information in WebRTC requires the implementation of robust approaches. Secure key exchange mechanisms, cryptographic and encryption protocols, secure authentication mechanisms, media integrity protection techniques, privacy-enhancing technologies, and network security measures collectively contribute to a secure and reliable WebRTC environment.

5. Case studies and implementation

5.1 Secure Multimedia Streaming via WebRTC

Secure multimedia streaming over WebRTC is a critical application that requires robust security measures to protect the confidentiality and integrity of transmitted content. Several case studies and implementations have focused on enhancing the security of multimedia streaming in WebRTC.

One approach involves using secure transport protocols, such as DTLS (Data Transport Layer Security), to encrypt the media stream in transit. By establishing a secure communication channel between participants, the confidentiality of the multimedia content is maintained and the risk of eavesdropping is reduced.

In addition, the implementation of media integrity protection techniques, such as cryptographic hash functions, can ensure the integrity of streaming content. By calculating and comparing the hash values, any tampering or alteration can be detected and the authenticity of the received multimedia data can be ensured.

In addition to encryption and integrity protection, access control mechanisms can be implemented to limit access to the multimedia stream. User authentication and authorization mechanisms, along with role-based access controls, help to ensure that only authorized individuals can access broadcast content.

5.2End-to-end encryption in WebRTC applications

End-to-End encryption is a powerful approach to ensure the privacy and security of multimedia information in WebRTC applications. This technique ensures that only the intended recipients can decode and access the multimedia content, even if the communication passes through intermediate servers. Implementing end-to-end encryption in WebRTC applications involves using cryptographic algorithms and secure key exchange mechanisms.

Participants generate unique encryption keys that are securely exchanged using protocols such as Diffie-Hellman or ECDH (Elliptic Curve Diffie-Hellman). These keys are then used for symmetric encryption and decryption of multimedia data.

Case studies have demonstrated the successful implementation of end-to-end encryption in WebRTC applications that enabling secure and private communication between participants. This approach provides strong protection against eavesdropping and unauthorized access to the multimedia stream and ensures that content remains confidential during communication.

5.3Privacy Techniques for WebRTC-Based Services

WebRTC-based services often involve the exchange of sensitive information, which raises privacy concerns for users. Various privacy-preserving techniques have been developed and implemented to address these concerns and protect the privacy of participants.

One of the techniques is to use selective encoding, where only certain parts of the multimedia content are encoded. For example, in a video conferencing application, the video stream can be selectively encrypted, while the audio stream remains unencrypted. This approach strikes a balance between privacy and usability. And ensures that communications remain functional while protecting sensitive information.

Another privacy-preserving technique is to use anonymous routing protocols, such as onion routing. These protocols ensure that the source and destination of the

Received: 21 March 2022
Accepted: 10 June 2022

communication remain anonymous as the traffic passes through a series of nodes. This protects the identity and location of participants and makes it difficult for unauthorized entities to track or identify them.

In addition, pseudonymization techniques can be used to replace identifying information with pseudonyms, further protecting participant privacy. By using pseudonyms, the real identity of users is hidden, privacy is increased and the risk of unauthorized identification is reduced.

In conclusion, case studies and implementations focusing on secure multimedia streaming, end-to-end encryption, and privacy-preserving techniques in WebRTC-based services have demonstrated successful approaches to enhance information security.

These implementations provide valuable insights into practical strategies for protecting multimedia content, ensuring privacy, and protecting the confidentiality and integrity of WebRTC communications.

6. Evaluation and comparative analysis

6.1 Security Criteria and Evaluation Frameworks

Security metrics and evaluation frameworks play an important role in evaluating the effectiveness and strength of security measures implemented in WebRTC applications.

These metrics provide a quantitative and qualitative assessment of security aspects and help organizations and developers to make informed decisions about the level of security achieved.

One of the common security measures is the confidentiality measure, which measures the degree of protection against unauthorized access to sensitive information.

This measure evaluates the effectiveness of encryption algorithms and access control mechanisms in protecting the confidentiality of multimedia data.

Another important criterion is the integrity criterion, which evaluates the ability to detect and prevent unauthorized changes or manipulation of multimedia content. By measuring the accuracy and effectiveness of integrity protection techniques, organizations can ensure the integrity of transmitted data.

Additionally, the availability metric evaluates the ability of the WebRTC system to remain operational and accessible to authorized users. It measures the effectiveness of network security measures, load balancing mechanisms, and denial of service mitigation strategies in ensuring uninterrupted multimedia communications.

Assessment frameworks, such as Common Criteria (CC) and National Institute of Standards and Technology (NIST) guidelines, provide a structured approach to assessing the security of WebRTC applications. These frameworks define a set of criteria, including security objectives, requirements, and assessment methods for assessing the overall security posture of a system.

6.2 Comparative analysis of security approaches

It is necessary to perform a comparative analysis of different security approaches in identifying their strengths, weaknesses and suitability for specific use cases in WebRTC applications.

Such analysis helps organizations and developers to make informed decisions about the most effective security measures to implement.

A comparative analysis can focus on different aspects of security, including encryption algorithms, authentication mechanisms, privacy-enhancing technologies, and media integrity protection techniques.

By evaluating the performance, efficiency, and security level of different approaches, organizations can choose the most appropriate measures to meet their specific security needs.

In addition, the analysis can consider factors such as computational overhead, interoperability, ease of implementation, and compatibility with existing infrastructure.

These factors affect the practicality and feasibility of implementing certain security approaches in real-world WebRTC environments.

Case studies and research papers comparing different security approaches provide valuable insights into the effectiveness and performance of different techniques. These studies evaluate security measures in terms of their ability to mitigate risks, protect sensitive information, and provide a secure and reliable multimedia communication platform.

As a result, it is very important to perform comparative evaluation and analysis of security measures and approaches in WebRTC applications to ensure strong information security.

Security metrics and evaluation frameworks help assess the effectiveness of security measures, while comparative analysis helps identify strengths and weaknesses of different approaches. Using these assessments, organizations can make informed decisions to increase the security of their WebRTC-based systems.

7. Future directions and emerging trends

7.1 Advances in Secure Key Exchange

Secure key exchange is a critical component of multimedia information security in WebRTC, and ongoing developments in this area have promising potential to enhance the security of WebRTC applications.

One emerging trend is the development of quantum-resistant key exchange mechanisms. As quantum computing poses a potential threat to traditional encryption algorithms, researchers are exploring new approaches such as post-quantum cryptography.

These techniques aim to develop encryption algorithms and key exchange protocols that can withstand quantum computer attacks and ensure the long-term security of WebRTC communications.

Another direction is to explore multi-factor authentication for key exchange. Traditional key exchange mechanisms rely on a single agent such as a password or cryptographic key. However, multi-factor authentication includes multiple factors such as biometrics, smart cards, or hardware tokens to enhance the security of the key exchange. This approach provides an additional layer of protection against unauthorized access and ensures the authenticity of participants [11].

7.2New encryption techniques

As technology evolves, so does the need for innovative encryption techniques to address emerging security challenges in WebRTC.

Several new encryption techniques show promise in enhancing the security of multimedia information. Homomorphic encryption is one such technique that performs calculations on encrypted data without the need for decryption.

This enables secure processing of sensitive multimedia information such as performing analytics or using machine learning algorithms, while maintaining privacy. Another emerging trend is the use of attribute-based encryption (ABE).

ABE allows access control to be defined based on specific characteristics of users, rather than traditional identity-based access control. This approach provides fine-grained access control and enables selective sharing of multimedia content based on user characteristics, such as role, location, or other specified criteria [12].

7.3Privacy-Aware WebRTC Implementations

Privacy has become a major concern in WebRTC applications, and future developments will focus on privacy-aware implementations to address these concerns.

One direction is to integrate different privacy techniques into WebRTC. Differential privacy aims to protect individuals' privacy by adding noise to aggregated data, thereby ensuring that individual data points are not easily identifiable.

By incorporating different privacy mechanisms into WebRTC implementations, it is possible to protect sensitive information while maintaining the usefulness of the data [13-18].

Another trend is the adoption of decentralized architectures and blockchain technology to increase privacy in WebRTC.

Decentralized systems reduce reliance on centralized servers and intermediaries, thereby minimizing the risk of data breaches and unauthorized access.

Blockchain technology, with its immutable and transparent nature, can provide secure and privacy-preserving records for WebRTC communications.

Additionally, emerging privacy-enhancing protocols, such as Multiparty Secure Computing (MPC), offer opportunities for privacy-aware WebRTC implementations.

MPC allows multiple parties to jointly compute a function while keeping their individual inputs private. This technique can be applied to WebRTC scenarios to enable secure collaboration and multimedia communication without endangering active privacy [19].

Consequently, future directions and emerging trends in WebRTC pave the way for advancements in multimedia information security.

Advances in secure key exchange, new encryption techniques, and privacy-aware implementations help develop more secure and privacy-preserving WebRTC applications.

By adopting these trends, organizations can ensure confidentiality, integrity, and privacy of multimedia communications (in an increasingly interconnected world) [20, 21].

8.Conclusion

Secure key exchange is a critical component of multimedia information security in WebRTC, and ongoing developments in this area have promising potential to enhance the security of WebRTC applications.

One emerging trend is the development of quantum-resistant key exchange mechanisms. As quantum computing poses a potential threat to traditional encryption algorithms, researchers are exploring new approaches such as post-quantum cryptography.

These techniques aim to develop encryption algorithms and key exchange protocols that can withstand quantum computer attacks and ensure the long-term security of WebRTC communications.

Another direction is to explore multi-factor authentication for key exchange. Traditional key exchange mechanisms rely on a single agent such as a password or cryptographic key. However, multi-factor authentication includes multiple factors such as biometrics, smart cards, or hardware tokens to enhance the security of the key exchange.

This approach provides an additional layer of protection against unauthorized access and ensures the authenticity of participants. As technology evolves, so does the need for innovative encryption techniques to address emerging security challenges in WebRTC.

Several new encryption techniques show promise in enhancing the security of multimedia information. Homomorphic encryption is one such technique that performs calculations on encrypted data without the need for decryption.

This enables secure processing of sensitive multimedia information such as performing analytics or using machine learning algorithms, while maintaining privacy.

Another emerging trend is the use of attribute-based encryption (ABE). ABE allows access control to be defined based on specific characteristics of users, rather than traditional identity-based access control.

This approach provides fine-grained access control and enables selective sharing of multimedia content based on user characteristics, such as role, location, or other specified criteria.

Privacy has become a major concern in WebRTC applications, and future developments will focus on privacy-aware implementations to address these concerns.

Additionally, emerging privacy-enhancing protocols, such as Multiparty Secure Computing (MPC), offer opportunities for privacy-aware WebRTC implementations.

MPC allows multiple parties to jointly compute a function while keeping their individual inputs private.

This technique can be applied to WebRTC scenarios to enable secure collaboration and multimedia communication without compromising privacy. Consequently, future directions and emerging trends in WebRTC pave the way for advancements in multimedia information security.

Advances in secure key exchange, new encryption techniques, and privacy-aware implementations help develop more secure and privacy-preserving WebRTC applications. By embracing these trends, organizations can ensure the confidentiality, integrity and privacy of multimedia communications in an increasingly interconnected world.

References

1. Bergkvist, D. C. Burnett, C. Jennings and A. Narayanan, "WebRTC 1. 0: Real-time Communications Between Browsers", W3C Editor's Draft 16, Jan. 2013.
2. B. Johnston and D. C. Burnett, "WebRTC: APIs and RTCWEB Protocols of the HTML5 Real-Time Web", Digital Codex, 2012.
3. Real-Time Communication in WEB-browsers (RTCWEB) IETF Working Group, [online] Available: http://tools.ietf.org/wg/rtcweb.
4. Zinah Nayyef, Sarah Amer and Hussain, "Peer to Peer Multimedia Real-Time Communication System based on WebRTC Technology", International Journal for the History of Engineering & Technology, vol. 2, no. 9, pp. 125-130, 201
5. Wajdi Elleuch, Models for multimedia conference between browsers based on WebRTC, pp. 279-284, 2013.
6. P Rodríguez, J Cerviño, I Trajkovska and J Salvachúa, "Advanced Videoconferencing Services Based on WebRTC", Proceeding of IADIS multi conference on computer science and information systems, 2013.
7. Why Does Your WebRTC Product Need a TURN Server?, [online] Available: https://www.callstats.io/blog/2017/10/26/turn-webrtc-products.
8. Julius Flohr, Ekaterina Volodina and Erwin P. Rathgeb, FSE-NG for managing real time media flows and SCTP data channel in WebRTC, 2018.
9. Edim Azom Emmanuel and Bakwa Dunka Dirting, "A Peer-To-Peer Architecture For Real-Time Communication Using WebRTC", 2017.
10. Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler"SIP: Session Initiation Protocol"RFC 3261 DOI 10.17487/RFC3261https://www.rfc-editor.org/info/rfc3261
11. Reschke, J."The 'Basic' HTTP Authentication Scheme"RFC 7617DOI 10.17487/RFC7617https://www.rfc-editor.org/info/rfc7617
12. Y. I. K. K. e. a. Ewan Leaver, "A Study of WebRTC Security," 26 July 2015. [Online]. Available: http://webrtcsecurity.github.io/. [Accessed 30 Nov 2015].

Received: 21 March 2022
Accepted: 10 June 2022

13. M. J. Werner, "WebRTC Security in the context of a DHT implementation," 2013.

14. E. Rescorla, "WebRTC Security Architecture," Internet-Draft – work in progress, March 7, 2015, 2015a.

15. K. Lee, J. Caverlee and S. Webb, "Uncovering social spammers: social honeypots+ machine learning," Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval, pp. 435--442, 2010.

16. Alcatel-Lucent, "WebRTC IMS Systems and WebRTC Proprietary Islands," technical white paper, June 2013

17. "WebRTC Security Concerns," WebRTC Solutions, 30 Jun 2014. [Online]. Available: http://blog.webrtcsolutions.com/webrtc-security-concerns/. [Accessed 09 Aug 2015].

18. P. Hancke, "Facebook Messenger likes WebRTC," 11 May 2015. [Online]. Available: https://webrtchacks.com/facebookwebrtc/.

19. "Anatomy of a WebRTC SDP," 2014. [Online]. Available: https://webrtchacks.com/sdp-anatomy/.

20. F. Culloca, "SDP - Glossary | MDN," 19 May 2015. [Online]. Available: https://developer.mozilla.org/enUS/docs/Glossary/SDP.

21. M. L. Mario Di Mauro, "A Decision Theory Based Tool for Detection of Encrypted WebRTC Traffic," in 18th International Conference on Intelligence in Next Generation Networks, 2015 .