International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Effect on Response Time of the Incorporated WebSocket Persistent Connection Feature in Qr Code Based Authentication

Ojal, Eddah Auma[1], Muhambe, Titus Mukisa[2], Obare, Erick Oteyo[3]

[1,3]Department of InformationTechnology, Maseno University, Kenya
[2]Department of Mathematics, Physics & Computer Science, Alupe University, Kenya

Abstract:
The need for secure authentication into systems has led to exploring of alternative secure authentication mechanism. QR code is deemed more secure and has thus increased in popularity, but implementation of QR code has downside of periodic polling; where client devices periodically poll the server to confirm whether the authentication was successful. Periodic polling contributes additional traffic, leading to increased response time, and often resulting to Denial of Service. The purpose of the study was to investigate the effect of implementing WebSocket feature to provide permanent connection between server and client in QR code authentication environment. Two experiments were setup where the control/reference experiment was used to determine baseline of response time for growing number of clients. It tested response time characteristics where periodic polling was at play. The treatment/ conceptual experiment implemented a persistent authenticated connection via WebSocket to eliminated periodic polling. The study hypothesized that a reduction in response time would be observed, when periodic polling in eliminated. The study applied experimental research design by simulating the control and treatment experiments, with an increasing number of clients in the OMNeT ++ simulator. Descriptive and inferential statistics were used to gather and evaluate data, comparing the performance of conceptual and reference models. It was observed that while the treatment/conceptual experiment demonstrated a considerable reduction of more than 80 percent, the response time in the control/reference experiment rose as the number of clients increased. With a standard variation of 22.99 ms, WebSocket persistent connection feature significantly lowered response time in QR code authentication. While the results look promising, the study recommended practical testing of this feature in real-environment to ascertain the model's ability to address the periodic polling challenge that negatively impacts on response times, which often leads to Denial of Service in Quick Response-based authentication**.**

Keywords: model; response time; QR code; dos; WebSocket; periodic polling

1.0 INTRODUCTION

1.1 Background

Critical business functions and processes have necessitated the implementation of computer networks within enterprises and connectivity to the internet. Transaction over the internet is the new normal the world over. To achieve effective and controlled access to enterprise systems, system authentication

methods are employed. Three conventional authentication techniques include: evidence of possession, knowledge, and traits [1]. Smart card serves as proof of possession, a password or pin code serves as knowledge, and human traits like thumb prints are used as characteristics [2]. Each conventional technique has its share of pros and cons; shareable passwords, recovery challenges, and device damage are a few of the traditional authentication challenges that can lead to security lapses or privacy violations [3] and [4]. The need for enhanced authentication has led to new approaches for authenticating into systems. These include multi-factor authentication which involves combination of elements of evidence of possession, knowledge and personal traits. Another secure approach that has gained popularity is the Quick Response (QR) code authentication.

1.2 QR-Code

QR code is designed as a two-dimensional square shaped matrix code that is differentiated by elements variation and dimension. Figure 1, shows parts of QR code, including format identifiers, version numbers, position markers, timing patterns, alignment markers, and data indicators.

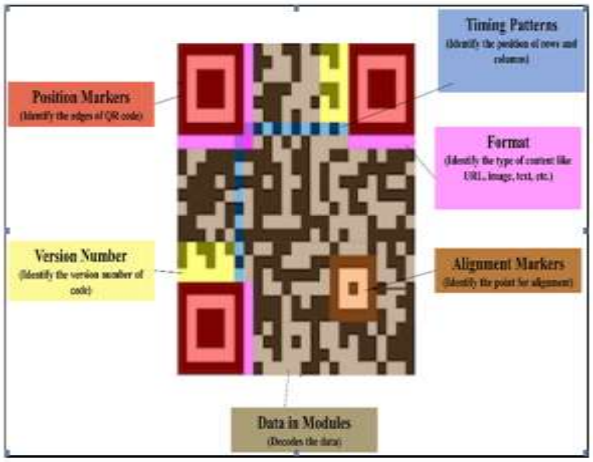

Figure 1: Structure of QR-Code (Source: Denso Wave, 2016)

In terms of structure and characteristics, the QR code size is set by the application version based on the amount of data, the type of characters, and the error recovery label [5]

The QR code is made up of square points called cells. The size of the cell is set by the resolution of the printer or scanner. There are 1–40 cells in each version. Starting with version 1, the cell has 21 cells horizontally and 21 cells vertically. In subsequent newer versions, 4 more cells are added progressively, until version 40, which has 177 cells horizontally and 177 cells vertically. Studies have shown that there is room for up to 7,089 digits (characters) in a single code [5],[6], [7] and [8].

The first QR code symbol is the position marker, which is a small square with alternating lighter and darker squares that serves as the code's position detection indicator [8].

1. *Position markers*: Known as position detection indicator, represented by a small square that combines lighter and a darker square. This indicates the QR code’s position and orientation.
2. *Timing pattern*: an interconnected patterns formed by the alternating sequence of dark and light elements. Its purpose if to indicate the size, number of rows and columns, and distortion in the Quick Response code.
3. *Version number*: Represenets the version number of the QR code.

4. *Format identifier*: a mask pattern number and the error correction level. This element is necessary for decrypting the content of Quick Response code e.g. URL, text, image, etc.
5. *Alignment marker*: serves as a detector for distortion. Accomplishes this function by identifying the alignment point in Quick Response code.
6. *Data indicator*: used to encrypt and decrypt data in QR code.

1.3 Traditional Utilization of QR-Code

Traditionally, industries utilize QR-Code for data storage, document management, product tracking, item identification, security, and authentication. The dots are transformed into characters or numbers by smartphones or QR scanners, which open a consistent resource location [9].

2.0 REVIEW OF RELATED LITERATURE

2.1 QR-Code Authentication

The Quick Response (QR) Codes authentication technique is categorized as proof of possession and is traditionally employed to encrypt and store data [10]. Its enhanced security features and successful use in industry have led to rapid and wide adoption as a more secure option for authenticating into systems in client/server and internet environments [11], [12]. Although the use of this technique presents a secure option against most authentication threats and attacks, it introduces periodic polling, where a client device that has requested for authentication from the server attempts to poll the server to confirm whether or not authentication was successful**.**

The time it takes for a client to ask for and get authentication from a server is known as response time, and it is expressed in milliseconds. According to [11], [12], [13], an average of 200 milliseconds is recommended.

2.2 Effects of Periodic Polling

In an environment with a growing number of clients, the periodic polling introduces extra traffic, directed to the server [14], [15], [16]. In QR-Code authentication in Client/server setup, the average response time is 200 milliseconds for a network environment with average load/traffic, but this may vary above or below depending on the network activities, load on the network and device capabilities [16]. Periodic polling leads to additional traffic that consumes bandwidth and increases server processing cycles and leads to longer response times for requested services. An increased response time has the potential to cause denial of service, especially in mission critical time sensitive services like voice and video calls.

2.3 Summary of Studies QR-Code Authentication

Several studies [17], [18], [19], [20] have researched the effects of periodic polling while other have attempted to control the periodic polling in QR-Code authentication environments. It has been established [21], found that in secure payment systems that implement QR codes for online transactions, the systems showed low processing time and high computational speed but were vulnerable to denial-of-service attacks due to the request-response pattern when multiple users request authentication. It has also been demonstrated [22] that using cloud-based cryptography to implement QR-code-based lightweight authentication protocols was a secure option but frequent traffic from clients in the form of periodic polling often overwhelmed the web servers causing denial of service.

Further [23], [24], [25], [26] in an attempt to prevent keyloggers and phishing attacks, QR codes are used for secure login on host machines but even though the approach is promising in terms of secure authentication, the client periodically sends polling requests to the web server to verify the device's

authentication attempt, but increased requests can overwhelm the server causing response time issues and DoS attacks.

Table 1, summarizes outcome of studies that have attempted to apply different approaches in the effort to understand the effects of these approaches on response time and the potential of the approach to lead to Denial of Service.

Table 1. Summary of the Related Studies

Study	Algorithm	Parameter	Finding
[24]	QR code with Http polling and Visual cryptography	Response time	Finally, it has low processing time and computational speed but is vulnerable to denial of service due to request /response pattern when many users request authentication
[23]	QR code with Http polling and cloud cryptography	Response time	Finally, the scheme is denial-of-service-vulnerable and has a long response time.
[26]	QR code with Http polling	Response time	Finally, the scheme is vulnerable to DoS and high latency. An absent user could give their device to a present user to authenticate.
[20]	QR code with Http polling and edge technology	Response time	Finally, there is increased response time resulting to Denial of Service
[17]	QR code with HTTP polling	Response time	Finally, the scheme is vulnerable to DoS and high latency.
[27]	QRP with HTTP polling	Response time	Finally, the scheme has high latency, denial of service, and server traffic with many requests.
[28]	QR code with Http polling	Response time	Periodic polling introduces traffic to the server increasing response time causing denial of service. Finally, the scheme's response time and server traffic are high, network congestion and latency
[18]	QR code with Web RealTime Communication (WebRTC)	Response time	According to analysis. Multiple incoming streams cause denial of service and server traffic.
[7]	QR code with HTTP polling	Response time	Due to server traffic, the scheme is not denial-of-service-proof.
[19]	QR code with HTTP	Response time	Verifying public key certificates

	polling		between the server and mobile device lead to increased response time and computing costs.

Studies [1], [2], [4], [15], [18], [22], [23], [28], [30], and [33] have shown that Although employing QR codes is secure, periodic polling problem increases response time resulting to Denial of Service attacks, which could severely impair mission-critical networks and destroy services.

3.0 METHODOLOGY

3.1 Research Design

The study adopted experimental research design where two experiments were setup; the control experiments and the treatment experiment were performed. To determine the response time baseline for conceptual QR code authentication, a control experiment is required to investigate the impact of eliminating periodic polling using WebSocket secure persistent connection.

The control experiment was used to establish the baseline while the treatment experiment was used to investigate the effects of eliminating periodic polling in QR-Code authenticated client server environments.

3.2 Experiment: Environment, Materials and Tools

3.2.1 Host Environment -Computer Specifications

An Intel Core i5 Dual Processor, 8GB of RAM, and an 80GB hard drive were needed for the study simulation. A host computer with a 1.2 GHz ANDROID 5.1 processor, 512MB RAM, and 4GB internal memory shared resources with the simulator. Android 6.0, Windows 10 64-bit, and an Intel Core i5 processor were employed in the study.

3.2.2 Simulator

Response time statistics and client information were analyzed using a Python and MySQL-programmed simulator, using OMNeT++ online simulator 6.0.1 for client/server functionality and QR code scanning.

3.3 Experiments Setup

Two experiments; Control and treatment experiments, modeled on the client server setup were conducted in an OMNeT ++ simulator. In both control and treatment experiments, there was no other data load apart from authentication and periodic polling data.

3.3.1 Control Experiment – Reference Model

In this experiment, the setup was the traditional client/server environment where QR-Code authentication was enabled with periodic polling at play. In the setup, the client initiates authentication, the server responds. The growing number of authenticating client devices was set to start with 10 devices and incremented automatically in steps 50, every 2ms up to a maximum of 1000. The increment simulated a growing number of devices in a client/server network environment. The Control experiment was necessary to establish the response time baseline for the traditional QR code authentication environment where periodic polling is at play.

3.3.2 Treatment Experiment (Conceptualized Model)

The effect on response time, of eliminating periodic polling in QR code authentication can be determined through an experiment that introduces secure persistent authenticated connection that elimin-

tes periodic polling.

In this setup, the conceptualized model incorporated a WebSocket (Figure 2) to provide a secure, authenticated and persistent connection between the client seeking authentication and the authenticating server. Data sharing without polling is made possible via the WebSocket standard, which establishes a persistent full-duplex connection between a client and server. Ten clients are added first and progressively in steps of fifty, up to 1000, every 2ms as a result of the OMNeT++ simulator initiating communication. The environment has no other load and server sends authentication information to clients over WebSocket connections. The HTTP protocol is upgraded to the WS protocol through a handshake.

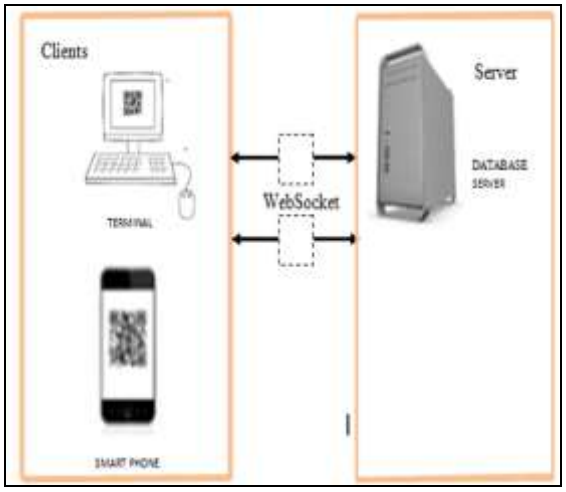

Figure 2. Conceptual Model

3.3.2.1 Persistent Authenticated Connection: Incorporation of Web Socket Feature

WebSocket is a persistent connection method that allows low-overhead, real-time client-server communication and two-way TCP link communication for real-time web applications [29]. The operation of incorporated WebSocket guaranties handshake between client and server while eliminating periodic polling. By providing a single socket for two-way, full-duplex communication, the WebSocket protocol builds reliable real-time web applications. It is appropriate for sophisticated real-time applications since it resolves HTTP issues and offers a browser-native socket. The WebSocket protocol facilitates communication over proxy servers and firewalls by allowing the simultaneous delivery of data, text, and binary frames over a connection. In comparison to HTTP, it eradicates periodic polling [30].

3.4 Data Collection, Analysis and Testing of Hypothesis

The study measured response time in control and treatment experiments using the OMNeT ++ simulation. A timer was incorporated to determine the overall time required for authentication and server response, and the results were saved in a simulation object. Excel sheets were used to collect and evaluate response time data from OMNeT++6.0.1.Response time observations were then analyzed.. To identify significant differences, averages and standard deviations were calculated and displayed on graphs. Descriptive and inferential statistics were used in the quantitative analysis, and tables were used for summaries. The study examined the reaction time of QR code-based authentication models using T-

tests for statistical comparison and published results in a variety of formats, such as tables, charts, graphs, and figures.

4.0 RESULTS AND DISCUSSION

4.1 Results for Control Experiment – Reference Model

The study used a reference model for expanding client/server systems to evaluate response time changes in QR code-based authentication. Although the average response time is 200 ms, there may be variances depending on device capabilities, network activity, and network congestion.

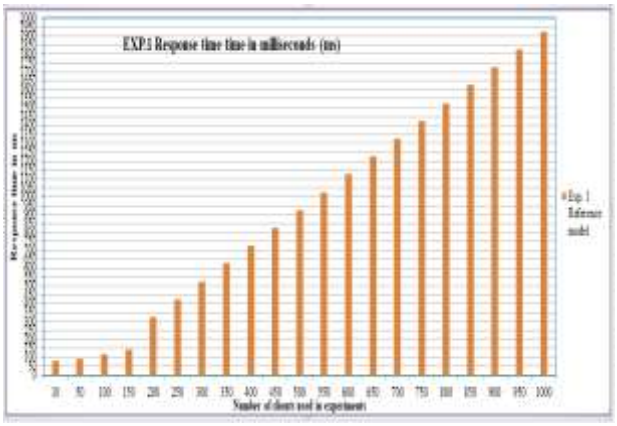

Figure 3: Results of Reference Model

The study shows the reference model's response time for QR code authentication, which uses periodic polling. For 100, 450 and 950 clients the response time is 119.88 ms, 825.32 ms and 1825.91 ms respectively, showing increase in response time. As the number of devices grows response time increases leading to Denial of Service. This was done without additional load. The results of control experiment are in agreement with studies [23], [24], [23], [7], [28], [17], [27], [18], [19] and [20].

4.2 Results for Treatment Experiment: Conceptual Model

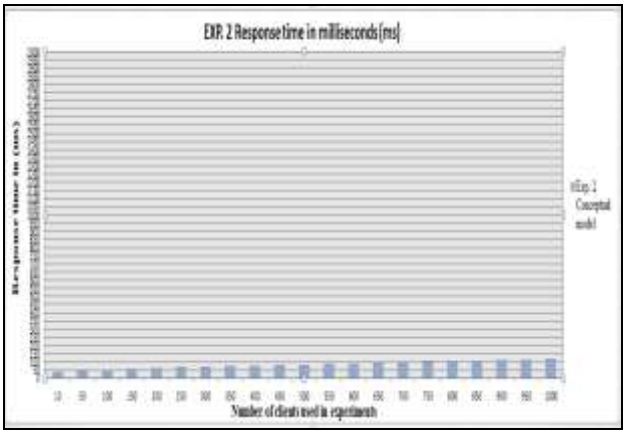

Figure 4: Results of Conceptual Model

Whereas studies [28], [17], [27], [18], [19] and [20] have shown that eliminating periodic polling may reduce server response time and eradicate Denial of Service when a considerably large number of clients poll the server at the same time. For 100, 450 and 950 clients the response time is 50.00 ms, 76.59 ms and 109.89 ms respectively; the environment has no other load. The results show a significant reduction in response time. The results seem to agree with the study [28], [17], [27], [18], [19], and [20] that says eliminating periodic polling may reduce response time and eradicate DoS.

4.3 Comparative Analysis of Control and Treatment experiment results

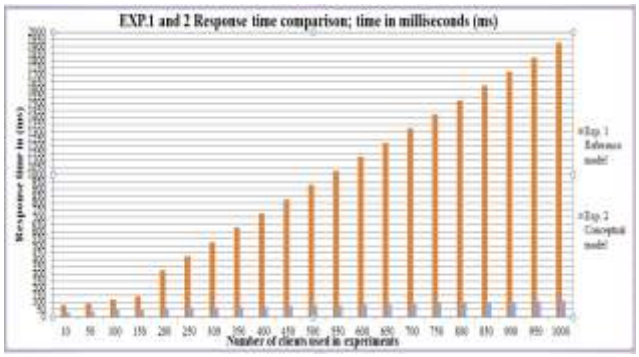

Figure 5: Comparative Analysis of Experiments

In a QR code-based authentication context, the conceptual model eliminates periodic polling, reduces response time and eradicated DoS as it incorporates WebSocket secure persistent connection[31]. Periodic polling in reference model increases response time resulting to DoS[32], [34], and [35]

5.0 CONCLUSIONS

The purpose of the study was to assess how adding WebSocket's permanent connection functionality to QR code-based authentication will affect expanding client/server setups. The study discovered flaws in basic password authentication techniques that result in security lapses. Although QR code authentication is secure, querying on a regular basis increase response time causing Denial of Service attacks. Two simulated networks were employed in the study: one using WebSocket and the other with periodic polling. When compared to the reference model, the conceptual model cut response time by 80%. By using WebSocket, frequent polling was removed, which decreased server response time and got rid of Denial of Service. The study which sort to eliminate periodic polling seems to show with elimination of periodic polling reduces response time and eradicates DoS. The alternative hypothesis is supported by the investigation, while the null hypothesis is rejected.

REFERENCES

1. Kaur, N. (2021, March). A study of biometric identification and verification system. In *2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)* (pp. 60-64). IEEE.
2. Kumar, S. (2019). A Review on Client-Server based applications and research opportunity. *International Journal of Recent Scientific Research*, *10*(7), 33857-3386.
3. Ramamoorthi, L. S., & Sarkar, D. (2020). Single Sign-On: A solution approach to address inefficiencies during sign-out process. *IEEE access*, *8*, 195675-195691.
4. Watanabe, Y., Suzuki, H., Naito, K., & Watanabe, A. (2019, October). Proposal for User Authentication Method Combining Random Number and Password. In *2019 IEEE 8th Global Conference on Consumer Electronics (GCCE)* (pp. 1129-1130). IEEE.
5. Bin, W., Hongjie, H., Xinwen, G., Yilin, X., & Yumeng, F. (2024, October). Design and Implementation of University Emergency Management System Based on QR Code Scanning and Mobile Terminal. In *2024 International Conference on Artificial Intelligence of Things and Systems (AIoTSys)* (pp. 1-5). IEEE.
6. Devi, P. J., Dutta, M. S., Damerakonda, M., Gutti, D. N., & Domakuntla, S. K. (2023, August). One Time QR-Code for Fake Product Identification. In *2023 5th International Conference on Inventive*

Research in Computing Applications (ICIRCA) (pp. 583-588). IEEE.

7. Kaarthik, K., Manibharathi, T., & Rakshith, D. (2022, May). QR Code based Shopping System. In *2022 International Conference on Applied Artificial Intelligence and Computing (ICAAIC)* (pp. 1005-1010).
8. Kant, S., & Ojha, S. P. (2022). An Improved User Interface for Enabling Smart Access Using Low-Cost QR-Based Systems at Various Points. In *Soft Computing: Theories and Applications: Proceedings of SoCTA 2021* (pp. 213-225). Singapore: Springer Nature Singapore.
9. Scanzio, S., Rosani, M., Scamuzzi, M., & Cena, G. (2024). QR codes: From a Survey of the State-of-the-Art to Executable eQR Codes for the Internet of Things. *IEEE Internet of Things Journal*.
10. Kumar, N., Jain, S., Shukla, M., & Lodha, S. (2022, June). Investigating users' perception, security awareness and cyber-hygiene behaviour concerning QR code as an attack vector. In *International conference on human-computer interaction* (pp. 506-513). Cham: Springer International Publishing.
11. Bhamidipati, V. S., & Wvs, R. S. (2022, November). A novel approach to ensure security and privacy while using qr code scanning in business applications. In *2022 Seventh International Conference on Parallel, Distributed and Grid Computing (PDGC)* (pp. 198-203).IEEE.
12. Sinha, S., Kaswan, S., Kumari, K., Kumar, A., Bisht, L., & Katiyar, S. (2024, March). A Novel Approach to Enhance Campus Lost and Found Services through Integration of QR Code with Personalized Item Registration. In *2024 International Conference on Trends in Quantum Computing and Emerging Business Technologies* (pp. 1-7). IEEE.
13. ALSaleem, B. O., & Alshoshan, A. I. (2021, March). Multi-Factor Authentication to Systems Login. In *2021 National Computing Colleges Conference (NCCC)* (pp. 1-4). IEEE.
14. Geetha, V., Gomathy, C. K., Kommuru, K., & Nallamsetty, P. (2024, July). Company employee profile using QR code. In *AIP Conference Proceedings* (Vol. 3028, No. 1). AIP Publishing.
15. Ismail, S., Alkawaz, M. H., & Kumar, A. E. (2021, April). Quick response code validation and phishing detection tool. In *2021 IEEE 11th IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE)* (pp. 261-266). IEEE.
16. Wang, R., Huang, L., Madden, K., & Wang, C. (2024, May). Enhancing QR Code System Security by Verifying the Scanner's Gripping Hand Biometric. In *Proceedings of the 17th ACM Conference on Security and Privacy in Wireless and Mobile Networks* (pp. 42-53).
17. Michelin, R. A., Zorzo, A. F., Campos, M. B., Neu, C. V., & Orozco, A. M. (2018, December). Smartphone as a biometric service for web authentication. *International Conference for Internet Technology and Secured Transactions (ICITST)* (pp. 405-408). IEEE.
18. Sharma, D. (2018, August). Response time based balancing of load in web server clusters. In *2018 7th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO)* (pp. 471-476). IEEE.
19. Taraka Rama Mokshagna Teja, M., & Praveen, K. (2022). Prevention of phishing attacks using qr code safe authentication. In *Inventive Computation and Information Technologies: Proceedings of ICICIT 2021* (pp. 361-372). Singapore: Springer Nature Singapore.
20. Yahya, Z., Kamarzaman, N. S., Azizan, N., Jusoh, Z., Isa, R., Shafazand, M. Y., ... & Mokhtaruddin, S. Z. S. (2019). A new academic certificate authentication using leading edge technology. In *Proceedings of the 2019 International Conference on E-commerce, E-Business and E-Government* (pp. 82-85).

21. Arif Hassan, M., Shukur, Z., & Kamrul Hasan, M. (2021). Enhancing Multi-factor User Authentication for Electronic Payments. In *Inventive Computation and Information Technologies* (pp. 869-882). Springer, Singapore.
22. Zhong, X., Xiong, L., & Xia, Z. (2021). A Secure Visual Secret Sharing Scheme with Authentication Based on QR Code. *Journal on Big Data*, *3*(2), 85.
23. Aciobanitei, I., Buhus, I. C., & Pura, M. L. (2020, May). Using cryptography in the cloud for lightweight authentication protocols based on QR codes. In *2020 IEEE 12th International Symposium on Applied Computational Intelligence and Informatics (SACI)* (pp. 000539-000542). IEEE.
24. Ahmad, L., Al-Sabha, R., & Al-Haj, A. (2021, March). Design and Implementation of a Secure QR Payment System Based on Visual Cryptography. In *2021 7th International Conference on Information Management (ICIM)* (pp. 40-44). IEEE.
25. Al-Ghaili, A. M., Kasim, H., Othman, M., & Hashim, W. (2020). QR code based authentication method for IoT applications using three security layers. *Telkomnika*, *18*(4), 2004-2011.
26. Allen, C., & Harfield, A. (2019, July). Authenticating physical location using QR codes and network latency. In *2019 14th International Joint Conference on Computer Science and Software Engineering (JCSSE)* (pp. 1-6). IEEE
27. Morales-Hernández, M., Morales-Jiménez, I., Osorio-Hernández, L. E., & Diaz-Sarmiento, B. (2021). Prototype of a web and mobile application for inventory management of a parts store using QR code. *Journal of Computational Systems and ICTs*, 7-19.
28. Khedr, W. I. (2018). Improved keylogging and shoulder-surfing resistant visual two-factor authentication protocol. *Journal of Information Security and Applications*, *39*, 41-57.
29. Murley, P., Ma, Z., Mason, J., Bailey, M., & Kharraz, A. (2021, April). Websocket adoption and the landscape of the real-time web. In *Proceedings of the Web Conference 2021* (pp. 1192-1203).
30. Nakamura, K., Inoue, T., Nishino, M., & Yasuda, N. (2021, December). Efficient Network Reliability Evaluation for Client-Server Model. In *2021 IEEE*
31. AlQahtani, A. A. S., El-Awadi, Z., & Min, M. (2021, October). A Survey on User Authentication Factors. In *2021 IEEE 12th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)* (pp. 0323-0328). IEEE.
32. Anakath, A. S., Rajakumar, S., & Ambika, S. (2019). Privacy preserving multi factor authentication using trust management. *Cluster Computing*, *22*(5), 10817-10823.
33. Bellavista, P., Dolci, A., & Giannelli, C. (2018, June). MANET-Oriented SDN: motivations, challenges, and a solution prototype. In *2018 IEEE 19th International Symposium on" A World of Wireless, Mobile and Multimedia Networks"(WoWMoM)* (pp. 14-22). IEEE.
34. Maestre, D. P. (2018). QRP: An improved secure authentication method using QR codes. *Universitat Oberta de Catalunya*, 1-11.
35. Zong, Y., Liu, S., Liu, X., Gao, S., Dai, X., & Gao, Z. (2022). Robust synchronized data acquisition for biometric authentication. *IEEE Transactions on Industrial Informatics*, *18*(12), 9072-9082