

**CSEC.472.600 – Authentication and Security Models**

**Spring Semester-2024**

**Homework2**

**A Comprehensive study of Multi-Factor Authentication (MFA) and Identity Access Management (IAM) within the Zero Trust Model**

Afra Mustafa

Jude Abdel Halim

Kassem Darawcha

Muaz Osman

Rana Alzakary

Dr. Wesam Almobaideen

April 30, 2024

Table of Contents

[Abstract 3](#_Toc165412853)

[1. Introduction 4](#_Toc165412854)

[1.1 Background Information 4](#_Toc165412855)

[1.2 Contribution 8](#_Toc165412856)

[1.3 Problem 8](#_Toc165412857)

[1.4 Motivation 9](#_Toc165412858)

[1.5 Methodology 9](#_Toc165412859)

[1.6 Structure 10](#_Toc165412860)

[2. Related work 11](#_Toc165412861)

[3. Comparative Analysis 16](#_Toc165412862)

[4. Observations 19](#_Toc165412863)

[Observation 1 19](#_Toc165412864)

[Observation 2 19](#_Toc165412865)

[Observation 3 20](#_Toc165412866)

[Observation 4 20](#_Toc165412867)

[Observation 5 20](#_Toc165412868)

[5. Recommendations 21](#_Toc165412869)

[Recommendation 1 21](#_Toc165412870)

[Recommendation 2 21](#_Toc165412871)

[Recommendation 3 22](#_Toc165412872)

[Recommendation 4 22](#_Toc165412873)

[Recommendation 5 23](#_Toc165412874)

[6. Proposal Motivation 23](#_Toc165412875)

[OpenID Connect Protocol (OIDC) 24](#_Toc165412876)

[7. OpenID Connect Protocol (OIDC) Attacks 28](#_Toc165412877)

[Replay Attacks 28](#_Toc165412878)

[ID Spoofing 30](#_Toc165412879)

[8. Proposed Solution Explanation 32](#_Toc165412880)

[Innovation Journey 35](#_Toc165412881)

[Results and Discussion 37](#_Toc165412882)

[Explanation of the Programs 37](#_Toc165412883)

[Client Program 37](#_Toc165412884)

[Server Program 38](#_Toc165412885)

[Pseudocode 40](#_Toc165412886)

[Expected Results  ( Enhancing Security Measure) 42](#_Toc165412887)

[Comparison with Yousra et al. (2023) 44](#_Toc165412888)

[Comparison with Ofleh (2018) 45](#_Toc165412889)

[9. Future Work 46](#_Toc165412890)

[10.Conclusion 47](#_Toc165412891)

[Members contribution 49](#_Toc165412892)

[References 50](#_Toc165412893)

# Abstract

This research paper explores the importance of identity access management (IAM) and multi-factor authentication (MFA) in contemporary cybersecurity frameworks for authentication by examining how they are integrated into the Zero Trust Model. This paper addresses an integrated literature review and a comparative analysis. Based on our findings and observations on the literature review papers, we will provide some recommendations and propose a solution to strengthen the authentication process and make it more secure in cloud computing within the zero-trust model. The main focus of our solution is the use of the OpenID Connect protocol (OIDC) and enhancing it by integrating multi-factor authentication and identity access management. The result of this paper is the implementation of a solution that mitigates replay attacks in the context of the OIDC protocol by replacing the single sign-on (SSO) with MFA and integrating it with IAM within the zero-trust model in cloud computing.

***Keywords****:* Authentication, Multi-Factor Authentication (MFA), Identity Access Management (IAM), Zero Trust Model, OpenID Connect protocol (OIDC), Cloud Computing.

# Introduction

## 1.1 Background Information

The role that authentication plays in cybersecurity has become critical considering the constantly evolving digital environment in which data is shared instantly and transactions take place on virtual platforms. Authentication is used to verify and confirm the identity of a person, system, or device that is trying to access a certain resource or service by verifying that the identity being claimed is legitimate and that the person is allowed to perform specific actions and access particular data (Kanade, 2023). Authentication provides access control by comparing a user's credentials with those stored in a data authentication server, which maintains the security of systems (Quest, 2024). There are several types of authentication: single-factor authentication (SFA), which involves only a user ID and password for access; then organizations have increasingly employed stronger authentication methods, such as two-factor authentication (2FA), which requires additional verification like a unique code sent to a mobile device or biometric data; and then multifactor authentication (MFA) takes this a step further by employing two or more factors for authentication to enhance security measures systems (Barney, Shacklett, & Rosencrance, 2023). Furthermore, as part of this paper, we will further explore concepts such as Zero trust model, multi-factor authentication (MFA), and identity access management (IAM), which play important roles in strengthening system security.

The Zero Trust model is a cybersecurity model that challenges the traditional concept of trusting networks based on the geography or identity of users. Instead, it considers that threats could arise both from outside and within the network at any moment. In a zero-trust model, every access request is thoroughly checked. This means people, devices, and apps must continually prove their trustworthiness before being permitted access to any resources (Shore et al., 2021). Zero Trust model core principles include explicitly verifying access, employing least-privileged access, and assuming a breach. The main objective of a Zero Trust architecture is to improve network security by minimizing the attack surface and lowering the risk of unauthorized access; hence, its approach focuses on dynamic access control, sensitive data encryption, and the idea of "never trust, always verify" (Zero Trust Model - Modern Security Architecture | Microsoft Security, n.d.).

An important security mechanism that is used in authentication is called multi-factor authentication (MFA), which means that a system needs a user to provide a combination of two or more credentials to confirm their identity in order to log in and access resources like apps, online accounts, and VPNs. This is an integrated approach to data and application security (CISA, 2022). Having multiple verification forms for MFA improves security because it prevents unauthorized access regardless of whether one of them is compromised, as it requests additional proof of identity from other categories. When authentic MFA is used, there might be a temporary passcode (possession) in addition to a password (knowledge) that is used, as opposed to a password security question combination that is inadequate (Cisco, 2024).

A key component of modern cybersecurity is identity and access management (IAM), which provides secure access to organizational resources such as apps, files, and data for employees working both on-site and remotely (Microsoft Security, 2024). Identity Access Management (IAM) is one of the key concepts that is important in maintaining systems, where it focuses on individuals’ access to digital resources and their potential applications, and it prevents attackers from accessing data while making sure that every individual only has the proper privileges required to do what they are tasked with (IBM, 2024). Accessibility to particular AWS resources can be restricted by access controls that can be centrally managed by IAM, which manages what users are authenticated and can sign in to access the resources and what users are authorized and have permission to use the resources (Amazon Web Services, 2024).

The existing vulnerabilities that we have within the OpenID Connect Protocol (OIDC), particularly, the replay attacks and Identity (ID) spoofing, which strengthen the calls for a multi-dimensional comprehensive plan for this issue. The first mitigation for the replay attacks we recommend that the ID token lifetime duration to be reduced from 120 minutes to a more secure period of 15 minutes, which will effectively reduce the large window within which the ID token can be misused by an attacker (CyberArk, 2024). The usage of MFA (Multi-Factor Authentication) will require the existing SSO service of the organization to be replaced by multi-factor service, which means that the users need to give multiple proofs to verify their identity when needed, therefore strengthening the authentication process and making unauthorized access significantly more difficult.

To further strengthen our solution, we proposed the implementation of IAM systems, augmented with OAuth2 functionalities, which will allow the third party in the user resources under secure and restricted conditions without exposing the user credentials which can greatly enhance the security (Ofleh, 2018). To protect against the replay attacks, we proposed the usage of a strong cryptographic function during the generation of the encrypted nonce and using the security authentication functions which is called “owner of()” that is related to the legitimate source of the token ID. By introducing the encrypted nonce to the OIDC request this will ensure that every request is unique and traceable; Therefore, the risk of the attacker replaying an intercepted tokens is eliminated (Yousra et al., 2023; Auth0, 2023).

The solution we proposed includes advanced security protocols, such as a reduced expiration time of the ID token, the generation of a cryptographic nonce, and a logging mechanism, in order to bring in tighter security during the process of authentication. This is quite different from the existing literature, where solutions focused on theoretical models or wider security enhancements without details in the practical implementation. For example, Yousra et al. (2023) proposed to use the blockchain technology within the protocol to add more reliability and privacy to the transmitted data. In our simulation the security measures have been taken inside the operational structure of the protocol. the expected results include larger control of the token validity, improved process of verification, and detailed traceability of the steps on authentication

## 1.2 Contribution

Considering the lack of articles addressing the intersections of multi-factor authentication (MFA) and identity access management (IAM) within the Zero Trust Model, our research group decided to combine these three fields for a comprehensive literature study. We conducted an extensive review and analysis of existing literature, frameworks, and best practices. Each team member took on the duty of reading and summarizing a certain paper, which contributed to our overall comprehension. We then combined our perspectives and worked together to examine, compare, and synthesize our results, where we identified the strengths and limitations of MFA and IAM implementations within the Zero Trust Model. This joint effort allowed us to present valuable observations, comparisons, and recommendations based on our findings. Then, we will propose a solution from the recommendations that is based on the observations and discuss the results, where we focus on mitigating the replay attacks within the OpenID Connect protocol by switching SSO to MFA, integrating IAM inside the Zero Trust Model, reducing ID token expiration duration, and adding a security authentication function and encrypted nonce to improve authentication. And finally, we will compare our proposed solution with the proposed work of the literature review that we have found, and future work for mitigating ID spoofing will also be added.

## 1.3 Problem

There is a problem that most organizations are dealing with is that they use Zero Trust Model frameworks that use only one authentication mechanism which is either identity access management (IAM) or multi-factor authentication (MFA) mechanisms for better authentication. It is hard to find Zero Trust Model frameworks that combine the two best authentication mechanisms to enhance authentication which are identity access management (IAM) and multi-factor authentication (MFA).

## 1.4 Motivation

This paper aims to address the issue of using only one authentication technique, either identity access management (IAM) or multi-factor authentication (MFA), inside the Zero Trust Model framework in cloud computing. To do this, we will combine both authentication techniques (IAM and MFA) inside the framework of the Zero Trust Model using a literature study that includes multiple articles to improve the authentication of the Zero Trust Model used in organizational settings.

## 1.5 Methodology

As part of the methodology for this article, multiple steps will be conducted. To begin with the initial step, we will conduct a literature review of seven papers written from 2021 to 2024 focusing on exploring existing Zero Trust Model frameworks, the methods and most useful practices of multifactor authentication (MFA), and identity access management (IAM) in order to adopt these strategies within a Zero Trust Model framework. The next step involves creating a tabular comparison of the nine papers that we compiled in the literature review, depending on several factors. In the last step, an observational assessment will be conducted, examining the comparison table and the literature review to identify areas that need to be improved. Based on these observations, recommendations will be developed that will target specific requirements and provide solutions to make advancements. And based on the recommendations, we will propose a solution design that focuses on mitigating the replay attack in order to enhnace authentication by integrating IAM and MFA within the OpenID Connect protocol in cloud computing inside the Zero Trust Model framework. Then finally, we will compare our proposed solution to the proposed work of the other literature review, emphasizing how our proposed solution is better at implementing authentication.

## 1.6 Structure

In this paper, we study some of the most important security components, including identity access management (IAM) and multi-factor authentication (MFA) within the Zero Trust Model. In Section 1, we provide an introduction to the main ideas, background information, our contribution, problem, motivation, explanation of how we went through analyzing the literature, and state our findings. Following that, Section 2 will provide a detailed analysis of various scholarly works. Moving forward, Section 3 will provide a comparative analysis of the papers. In Section 4, we will discuss what we observed and gathered from the literature review; in Section 5, we will recommend some ideas based on what we observed; in Section 6, we will discuss what motivated our proposal; in Section 7, we will explain the OIDC attacks; in section 8, we will propose and implement a solution that was recommended to improve the authentication process; in Section 9, we will discuss our future work and how we can expand our solution; and finally, in Section 10, we will summarize our observations and recommendations and conclude what we found and implemented.

# Related work

The paper's main focus which is written by Nguyen et al. (2023) talks about the implementation of information security management systems (ISMS) in South Korea using the authentication mechanism which is multi-factor authentication and how to integrate it with the Zero Trust Model. The traditional security models are found to be inefficient against cyber threats so Nguyen suggests that a paradigm shift should be used in information security that relies on multi-factor authentication in the Zero Trust Model. In order to offer a solution to the discussed problem of the South Korean case, the paper offers an invented framework which is called "Zero Trust K-ISMS Framework" where this framework is a method that integrates Zero Trust principles that use multi-factor authentication in order to increase the measures of security within the Korean context (Nguyen et al., 2023).

The Zero Trust Model's paper by Kovacevic et al. (2024) explains in detail how to establish cloud service security using identity access management (IAM) in the Zero Trust Model where the author starts by introducing the concept of edge computing, which aims to lower latency and improve network access efficiency by relocating data centers closer to the data sources they need to access and this concept helps in mitigating the security threats that are in the micro-cloud environment. The main elements that are used in the upgraded Zero Trust Model are multi-factor authentication, certificate management that is used for device authentication, and attribute-based policies for access control that use the Extensible Access Control Markup Language (XACML) to control access to resources that depend on the features of the subject, object, and environment. Also, Kovacevic divides the use of the authentication methods for the upgraded Zero Trust Model into two parts where the first part is the authentication for users, which proposes multi-factor authentication, the second part is the authentication for devices, which proposes SSL authentication, and for both users and devices, they have digital certificates controlled by a Public Key Infrastructure (PKI) in order to provide secure authentication.

The main focus in the research paper by Daah et al. (2024) is some cybersecurity developments in financial institutions and the integration of the zero trust model in the blockchain technology, where they discuss some key components and mechanisms that have an important role in strengthening security measures, like multi-factor authentication (MFA) and identity access management (IAM). Before the deployment of Zero Trust standards, the device and network security framework relied on a perimeter-based protection strategy that focused on securing and protecting network boundaries instead of the internal security of the network. After applying MFA and IAM in the zero trust model, the authentication process starts when bank customers input their credentials, which are then checked in the bank's internal database records using more than one form of verification. If the login details are checked as valid, the system will generate session tokens, which gives them access to the bank application, on the other hand if the verification decline, customers will not have access. Therefore, integrating these mechanisms in the Zero Trust model and using blockchain technology will result in a stronger defense against internal and external security risks, which provides financial institutions with a comprehensive approach to protect sensitive data and transactions (Daah, Qureshi, Awan, & Konur, 2024).

Zero Trust Security's paper by Garbis and Chapman (2021) takes a deeper look into how IAM operates as one of the core components of information security, more precisely within the realms of the Zero Trust security model. They highlight the role of IAM in ensuring the right people have the correct access at the right time. Identity is the foundation of Zero Trust and advocates for the deployment of an identity-centric security approach across complex, multi-system landscapes of modern enterprises. This presents the progress of identity management systems, the development of Identity-as-a-Service (IDaaS), and, at the same time, the requirement to handle identity life cycles needed for security and compliance matters. Moreover, the research also considers identity governance as it is in the Zero Trust environment and its components on process authentication and authorization. Thus, according to Garbis and Chapman, Zero Trust and IAM are no longer just technical pursuits but, in fact, organizational imperatives that need integration to better both security and efficiency meaning a holistic approach to protect digital assets in a connected world (Gabris & Chapman, 2021).

The paper by Chen and Sun (2021) proposes a multi-level intelligent multi-factor authentication scheme for power mobile Internet services under the zero-trust architecture. It addresses the rising complexity and security concerns caused by unclear network boundaries and diverse access points in the smart city environment. By integrating dynamic and static authentication approaches, the strategy discussed intends to improve the flexibility, accuracy, and security of multifactor authentication. One of the key components of the proposed method is User-Terminal-Application Security Binding, which involves setting up user and terminal identities, binding them together, and controlling application services depending on security levels. Also, Static Authentication which is the initial verification of a user's identification by applying current static authentication methods such as passwords, facial recognition, and smart cards. And last, dynamic authentication is based on the Trust Evaluation Value which assesses the security and legality of user access. The scheme implements a two-way dynamic authentication structure between the user and the server to provide safe communication and access control. The security analysis mentioned in the paper is carried out using SVO logic, which demonstrates the scheme's robustness to multiple attacks. Furthermore, the time cost of the encryption step is examined, and a comparison to alternative authentication techniques are examined (Chen et al., 2021).

This paper by Iordache et al. (2022), is a journal article concerning the improvement of cyber protection within public institutions through multi-factor authentication with biometrics. The paper emphasizes the importance of utilizing biometric data as a standard practice to strengthen security measures. It points out, for example, the Zero Trust Model architecture approach, which lies in attesting every user and device trying to connect to the network from wherever they might be located. Together, these can be very effective at bringing down the risk of access from unauthorized sources and failure in data security. This paper would highly recommend the installation of complex security standards in public institutions to protect the relevant information (Iordache et al., 2022).

As I am talking about the Zero Trust Model framework, I found an article that integrates multi-factor authentication and intrusion detection to develop security in cloud computing environments. Said et .al's paper emphasizes how important it is to pay attention to certain factors in making multi-factor authentication (MFA) systems secure. These factors include how long the authentication data is, how long it's valid for, and checking if the entered information matches stored credentials. The paper talks about different processes that boost MFA security, mainly focusing on Authentication and Intrusion Detection. These processes help make sure that only the right people can access the application or system, making everything safer. Said et al. (2022) describe how the multi-factor authentication (MFA) process involves several steps to confirm users' identities before allowing access to the application. I analyzed it and found it contains passwords, security questions, and verification messages within the Zero Trust Model framework. For intrusion detection, Said et .al attached a Figure that explains the roadmap of the intursion detection, as I understood from the figure there are several steps like inspecting suspicious activities for unauthorized users and checking the length and validity of authentication data and comparing entered values with stored credentials, and contribute to the effectiveness of the intrusion detection system. I saw some Other metrics such as detection rate (DR), false positive (FP) alarm rates, and false negative (FN) alarm rates are used to evaluate the performance of the security measures.

Omer's paper is a thesis talking about the OpenID Connect (OIDC) standard, which is a widely adopted and fastest-growing Identity and Access Management (IAM) protocol, It discusses the need for secure and streamlined authentication mechanisms in systems. The OIDC standard provides recommendations for fundamental IAM challenges, such as user identifiable information, consent, and data access monitoring. It also discusses the importance of balancing security and performance for users through single-sign-on experiences. The paper highlights the distributed nature of identity ownership and verification, as well as the potential risks associated with different authentication mechanisms. Lastly, it recommends some proposals to improve this protocol, like the use of OpenID certification and two-factor authentication (2FA). The thesis focuses on the important role of successful and robust IAM implementations in today's organizations and the need for strategic decision-making in this complex landscape (Ofleh,2018).

The paper by (Yousra et al., 2023) talks about a proposed model for OpenID Connect (OIDC) based on blockchain technology, aiming to enhance data security, integrity, and privacy protection. The system uses the Ethereum blockchain and non-fungible token (NFT) standard to securely record OIDC sensitive parameters. It includes smart contract functions, which require a fee in the form of Gas to complete transactions. The proposed system also deals with user tracking, integrity, availability, revocation, and delegation. The performance evaluation of the system indicates that it does not affect user experience and the performance of existing OIDC-based systems. The paper also, includes a security analysis of the proposed system using formal and informal methods, as well as a comparative analysis of the proposed Blockchain-based OIDC approach and other similar systems. For the formal security analysis they used tools like AVISPA and Scyther to verify the security of the system.But the informal analysis involves discussing the security and privacy threats of the proposed system, as well as a comparative analysis with other similar systems.

# Comparative Analysis

#### **Table 1**

*Security Factors Comparison Table*

|  |  |  |  |
| --- | --- | --- | --- |
| **Paper** | **Focus Area** | **MFA** | **IAM** |
| **Said et al. (2022)** | -Focuses on integration of intrusion detection with MFA for cloud computing environments within the Zero Trust Model framework. | - Emphasizes the importance of authentication factor length and validity periods.  - Processes such as comparing entered factors with stored credentials highlighted. | Does not show a big focus on this factor. |
| **Iordache et al. (2022)** | -Focuses on Zero Trust Model that utilizes biometric data and its implications for security in public institutions. | - Urges the use of biometric data for MFA. | Does not show a big focus on this factor. |
| **Chen and Sun (2021)** | -Integrates dynamic and static authentication approaches within the Zero Trust framework for improved security in smart city environments. | - Proposes a multi-level intelligent MFA scheme.  -Static and dynamic authentication approaches integrated. | Does not show a big focus on this factor. |
| **Garbis and Chapman (2021)** | -Focuses on organizational imperatives of integrating Zero Trust Model and IAM for security and efficiency. | Does not show a big focus on this factor. | - Identity governance and process authentication emphasized. |
| **Daah et al. (2024)** | -Integrates MFA and IAM within the Zero Trust Model with blockchain technology for enhanced security in financial institutions. | - MFA with two or more forms of verification suggested.  - Integrates MFA and IAM within Zero Trust model using blockchain technology. | - IAM standards impose strong access control based on roles. |
| **Kovacevic et al. (2024)** | -Emphasizes the Zero Trust framework with the use of edge computing and attribute-based access control policies for enhanced security in cloud environments. | - Multi-factor authentication and certificate management discussed.  - SSL authentication and digital certificates usage discussed. | - Attribute-based policies for access control highlighted.  - Extensible Access Control Markup Language (XACML) for access control based on attributes. |
| **Nguyen et al. (2023)** | -Focuses on the adoption of MFA practices and the Zero Trust Model in the South Korean context. | - MFA implementation within the Zero Trust Model focused on. | Does not show a big focus on this factor. |
| **(Ofleh,2018)** | The exploration of the OpenID Connect (OIDC) standard and its role in the future of federated Identity and Access Management. | It talks about how OpenID Connect and Two-Factor Authentication (2FA) can strengthen authentication protocols. | talks about how OpenID Connect standard resolve the problem of IAM across networks. |
| **(Yousra et al., 2023)** | Talks about multiple proposals and models for addressing privacy and security concerns related to the OpenID Connect protocol in the context of blockchain technology. | Does not show a big focus on this factor. | Does not show a big focus on this factor. |

# Observations

## Observation 1

To Start observing Garbis and Chapman (2021) paper we need to look at what they mentioned, they said integrating the Zero Trust Model with Identity and Access Management (IAM) can make corporations more secure and much faster like it will be efficient . In contrast, if we are observing Daah et al. (2024) paper we can see it puts two things in one model, IAM and multifactor authentication within the Zero Trust Model.

## Observation 2

Now for Kovacevic et .al (2024) paper we can look that they mentioned cloud environments, and how they can make it better, but how? they said the Zero Trust framework could make a real difference by using edge computing and attribute-based access control mechanisms. They also look at multifactor authentication, explaining SSL authentication, managing certificates, and how digital certificates help adding some security. Nguyen et .al paper, on the other hand, Their paper concentrates on implementing the Zero Trust Model and multifactor authentication (MFA) protocols in South Korea how this contry use MFA within the Zero Trust Model.

## Observation 3

As we mentioned in the summary Said’s paper mentioned the cloud computing, this cloud can have very important additions, we can have an observation for Said et. al(2022) paper that could be talking about combining intrusion detection with multifactor authentication (MFA) for cloud computing in the Zero Trust Model. They say that how long and valid authentication data is really matters in MFA. They also talk about checking if the entered data matches stored credentials to boost security.

## Observation 4

Iordache et al. (2022) look into using the Zero Trust Model in public institutions, adding biometric data for stronger security. I saw and observed that they suggest that using biometric data is important for multifactor authentication. Similarly, Chen and Sun (2021) propose a multi-level intelligent multifactor authentication (MFA) system within the Zero Trust framework for improved security in smart city environments, combining dynamic and static authentication methods.

## Observation 5

We observed that the papers by Omer discuss the openid, the authentication mechanisms in systems, and the challenges faced with IAM. At the same time, it recommends some ideas to overcome these challenges, like OpenID certification and two-factor authentication (2FA) (Ofleh, 2018). On the other hand, the paper by Yousra et al. (2023) also talks about openids based on blockchain technology, recommends some solutions to the vulnerabilities seen in this protocol, and suggests some metadata extension methods that should be used in the protocol. Both papers serve the same purpose, which is to improve the authenticity and security of the open protocol.

# Recommendations

## Recommendation 1

Based on observation 1, to address the oversight identified by Garbis and Chapman (2021) and following the insights from Daah et al. (2024), multi factor authentication (MFA) should be offered by the Identity and Access Management (IAM) under the banner of the Zero Trust Model, which should be based on blockchain. This is to cover the need for at least two different check methods in MFA, across the board for all organizations, with significant security improvements contributed by another layer of defense from unauthorized access. These help organizations to fortify their security frameworks, providing strong protection within the Zero Trust Model.

## Recommendation 2

Based on observation 2, the recommendation is to extend the adoption of the Zero Trust framework by incorporating edge computing, and attribute-based access control in cloud environments. This calls for the need to integrate multi factor authentication (MFA) practices into the security domain of the organization, including, among others, certificate management, SSL authentication, and digital certificates. Furthermore, it may even be said that the model adopted by Nguyen et al. (2023) in their research is also to be used by future scholars, one regarding the implementation of MFA within the Zero Trust Model in the South Korean environment. This recommendation specifies that holistic security in technological advancements, like edge computing, needs to reinforce the cloud security of those advancements with multi-factor authentication methods that are solid and enhanced certificate management (for MFA) to solidify further that practice of enhanced strength within the Zero Trust model.

## Recommendation 3

Based on observation 3, the control of security protocols by users will be enhanced in cloud computing environments through the incorporation of MFA into the IDS under the Zero Trust Model framework. Emphasis should be laid on the most critical aspects of the length of authentication factors and their validity periods in the MFA processes. The approach will be embedded with strong procedures, while the authenticity of entered factors against stored credentials is tested. These, when established, are able to significantly improve the security and integrity of the realization of both strong and dynamically strong-responsive access controls in cloud-based systems within the OpenID Connect Protocol. This clearly points out the need for a sound, detailed, and layered security mechanism that will guard against advanced threats taking place in the cloud computing scenarios.

## Recommendation 4

Based on observation 4, it is recommended to use biometric data as one of the important factors in multi factor authentication (MFA) in only the most sensitive places of the Zero Trust Model, like public administration or smart city environments. This is where dynamic methods are brought together with static authentication methods to create a multi-level, smart MFA. Implementation of such a system would enhance the current security state by ensuring that the authentication processes are more resilient to unauthorized access and much better in ensuring that the identity of the user is verified. The approach not only leverages the intrinsic identity of biometric data but also conforms to "never trust; always verify" under the Zero Trust model, further incorporating a resilient, adjustable, and strong layer of security from mutating cyber threats.

## Recommendation 5

Based on observation 5, both papers focus on finding ways to enhance open protocol security. Discussing security concerns and talking about the attacks, they recommend using OpenID certification, two-factor authentication (2FA), \_burn(tokened), token(URI), transferFrom(add\_from,add\_to,tokenid), and lastly ownerOf(tokenid\_), which are functions that serve as a mitigation against many attacks that compromise the systems security. These mitigations deal with various attacks like replay attacks, ID spoofing, token and key confusion.

# Proposal Motivation

After observing the issues that are present in the literature review papers that we have summarized, we have decided to propose a solution based on observation 5. The solution that we will propose will be from recommendation 5 that we have discussed since we noticed that authentication is not that secure in cloud computing within the Zero Trust Model framework, where the OpenID Connect protocol needs to be enhanced by changing authentication mechanisms and also integrating additional mechanisms such as MFA and IAM that we discussed previously, adding attributes, and applying functions related to the OpenID Connect protocol in order to focus on enhancing the security and performance of authentication within the Zero Trust Model framework.

## OpenID Connect Protocol (OIDC)

OpenID Connect Protocol, or OIDC, is an identity authentication protocol layer that is built on top of open authorization OAuth 2.0, which is a technological standard that enables users to be authorized on two unrelatable applications and let them log in on one of them without providing their confidential data like passwords on the other one (Okta, 2024). OIDC standardizes the authentication process and facilitates the method by which users' identities are verified; it enables third-party applications to verify users' identities and obtain basic user profile information when users sign up for digital services. In addition, OIDC provides single-sign-on (SSO), where organizations employ a secure identity and access management (IAM) system to authenticate identities and then transfer that authentication to other applications using OIDC, which means that users just have to sign in once with the same credentials in order to use different applications (Microsoft, 2024).

OIDC consists of six essential elements, that can be summarized in the table below:

#### **Table 2**

*OpenID Connect protocol elements Table*

|  |  |
| --- | --- |
| **Element** | **Definition** |
| Authentication | The procedure used to confirm that a user is who they claim to be. |
| Client | Software or applications that require tokens in order to access resources or verify a user's identity. |
| Relying Party | Applications that validate users' identities by using OpenID providers. |
| Identity Tokens | Tokens that hold identification data, such as the user's identifier, the result of the authentication process, and details about the exact process and timing of the user's authentication. |
| OpenID Providers | Applications that a user has previously established an account with. They are responsible to authenticate the user's identity and provide the relying party with this information. |
| Users | Individuals or services that want to access an application without setting up a new account or entering their credentials. |

Below is the process for the OpenID Connect protocol, which enables secure authentication and identity verification to allow users to access a particular application by signing in to another (OpenID Foundation, 2023).

1. It begins when a user opens a browser and navigates to a website or web application.
2. The end user signs in and inputs their credentials (username and password).
3. A request is sent to the OpenID Provider (OP) by the relying party (RP).
4. The OpenID Provider (OP) validates user credentials, authenticates, and authorizes the user.
5. The OpenID Provider (OP) replies to the relying party (RP) with both an identity token and often an access token.
6. The user device receives a request from the relying party (RP) that includes the access token.
7. Claims regarding the end-user are returned by the UserInfo endpoint.
8. Based on the information stored in the access token and relying party (RP), the user is granted access.

A diagram of a system

Description automatically generated

***Figure1 .*** OpenID Connect Authorization flow

# OpenID Connect Protocol (OIDC) Attacks

## Replay Attacks

A replay attack takes place when an unauthorized party intercepts a private communication, retransmits it, and strategically delays or resends it to manipulate the recipient into taking actions beneficial to the attacker. In the context of OpenID Connect (OIDC), a token replay attack occurs when an attacker intercepts a legitimate ID token and then reuses it to gain unauthorized access to resources. This type of attack exploits vulnerabilities in the token validation process, allowing attackers to exploit weaknesses in token validation mechanisms (Yousra et al., 2023).

The interception of tokens typically happens in different ways, such as eavesdropping on network traffic or compromising the user's device where the token is temporarily stored (Yousra et al., 2023). Once the attacker gets the intercepted ID token, he can replay it by submitting the same token to the relying party (RP) or Service Provider (SP) many times. Since the token contains authentication information, the RP might accept it as valid and grant access to the attacker, assuming it to be a legitimate user (Heilman et al., 2024).

One key element contributing to OIDC's vulnerability to token replay attacks is the comparatively long expiration duration of the ID token, which is typically set to 120 minutes . This fixed expiration time adds limitations on session duration and increases the window of opportunity for attackers to carry out token replay attacks. Additionally, it may lead to the accidental use of old tokens beyond their expiration time, posing security risks if RPs fail to properly validate token expiration (IBM Cloud Application Performance Management, Private 8.1.4, n.d.).

A diagram of a security system

Description automatically generated

***Figure2 .*** Diagram of the attack

OpenID includes a parameter called openid.response\_nonce, which contains a timestamp concatenated with a random string, but it does not include an expiration time. Which means that the Service Provider (SP) can decide on its own how long it accepts such a token (Han & Lee, 2023). Notably, according to three different resources mentioned above, OpenID is vulnerable to token replay attacks. This highlights the importance of addressing this vulnerability as an important issue that requires attention and mitigation as the consequences of this attack are significant.

## ID Spoofing

ID spoofing (IDS) attack targets an OpenID protocol vulnerability related to identity authentication in Single Sign-On (SSO) systems. This type of attack falls in Category B, where it makes manipulation of information related to identity that is contained within the SSO token. The following elaborates on how the IDS attack works:

1. **Initiation of Login Attempt:** The attacker will initiate a login process on a Service Provider (SP) using a malicious Identity Provider (IdP) controlled by the attacker IdP.
2. **Malicious Ticket Generation:** The attacker IdP will generate an SSO token that supposedly represents an identity managed by a legitimate and honest IdP for example, Google. Still this fraudulent token is signed with the key that belongs to the attacker IdP.
3. **Token Submission:** If the Service Provider accepts this malicious crafted SSO token, the attacker gains unauthorized access to accounts that are normally managed under other legitimate IdPs.

A diagram of a system

Description automatically generated

***Figure3 .*** Diagram of the attack

This attack is meant to exploit the trust clients put in the issuer claims inside ID Tokens. Strict verification of these claims needs to be carried out against known, trusted issuer values that were obtained during the discovery phase (Mainka, Mladenov, & Schwenk, 2014).

# Proposed Solution Explanation

Based on the vulnerabilities that we are facing in the replay attack within the OpenID Connect Protocol, we are going to propose a solution to mitigate the reply attack within the OpenID Connect Protocol. In order to enhance authentication, we are going to divide the solution sections into four parts.

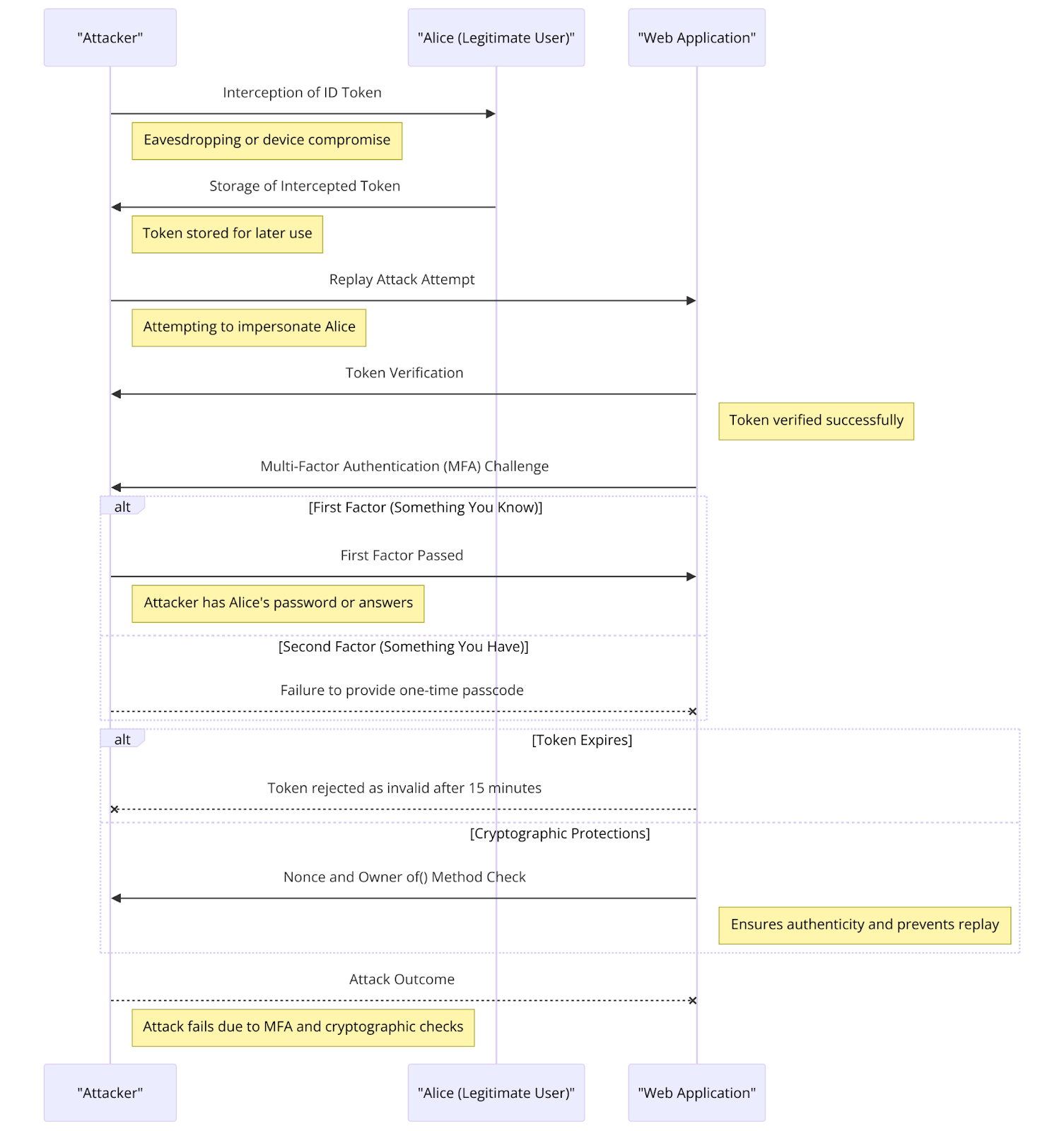
The first part will discuss the need to switch from single sign-on (SSO) to multi-factor authentication (MFA) within the OpenID Connect Protocol since single sign-on possibly allows attackers to gain unauthorized access and exploit weaknesses in token validation by only requiring single authentication, while multi-factor authentication makes the user use two or more authentication process whether if they are privileged to gain access or not (KOSE et al., 2020).

The second part will focus on integrating identity access management (IAM) within the OpenID Connect Protocol by extending it using the OAuth2 feature which is an authorization mechanism that allows third-party a limited to gain restricted access to user's resources in a secure way without the need to reveal their credentials in which this way the users grant permission to authorized third party through the use of ID tokens so this does not allow any other users to gain unauthorized access to the user’s resources in which it will increase the security of authentication (Ofleh, 2018).

The third part will cover reducing the ID token expiration time within the OpenID Connect Protocol, as discussed previously, the OpenID Connect Protocol vulnerability is that the ID token has a long expiration duration, which is typically set to 120 minutes, and this allows the systems to encounter replay attacks easily (CyberArk, 2024). In order to solve this, we will change the expiration duration of the ID token from 120 minutes to 15 minutes.

The final part will examine the use of cryptographic functions for encrypted nonce generation and the using a security authentication functions for ID token within the OpenID Connect Protocol, where encrypted nonce should be applied in the OpenID Connect request which is an integer unique number that ensures that the request is from an authentic user (Auth0, 2023). Using the “owner of()” method should also be applied which checks where the source of the ID token came from and whether it’s from the authentic and intended user or not so it prevents replay attacks (Yousra et al., 2023).

The following flowchart is an example of how an attacker couldn’t attack the web service by performing a replay attack because our proposed solutions prevented the replay attack.

****

***Figure4 .*** Diagram of the Replay Attack Mitigtion

# Innovation Journey

Now, we are going to discuss the difference between the proposed idea and other in the literature for each solution that we have proposed in order to enhance the security of the authentication.

For the first solution part, the paper that was written by Fett et al. (2017) have proposed the use of single sign-on within the OpenID Connect Protocol in order to maintain the performance of the network but it is not that much secure in terms of the authentication security mechanism since it only uses one form of authentication where the attacker could perform the replay attack and possibly bypass it. However, the proposed solution that we have proposed is to switch from SSO to MFA within the OpenID Connect Protocol so that two or more authentication forms will be used in order to authenticate the users and this will increase the security of authentication.

For the second solution part, Said et al. (2022) have used one mechanism, which is multi-factor authentication, in the OpenID Connect Protocol within the Zero Trust Model framework and switching from SSO to MFA within the OpenID Connect Protocol is not compatibility secure for authentication because using only the MFA mechanism without adding another authentication mechanism will reduce the authentication security and performance. While the proposed solution that we have proposed is to integrate another authentication mechanism with MFA in the OpenID Connect Protocol within the Zero Trust Model framework, which is identity access management, and it should be extended using the OAuth2 feature within the OpenID Connect Protocol for stronger authentication.

For the third solution part, the IBM paper has set a long expiration duration for the ID token, which is set to 120 minutes within the OpenID Connect Protocol, which is vulnerable to replay attacks (IBM Cloud Application Performance Management, Private 8.1.4, n.d.). However, the proposed solution that we have proposed is to reduce the expiration duration of the ID token to 15 minutes within the OpenID Connect Protocol to mitigate the replay attack and strengthen the authentication within the OpenID Connect Protocol in the Zero Trust Model.

For the fourth solution part, Heilman et al. (2024) paper illustrates that the attackers would eavesdrop on a legitimate user to get the ID token and perform reply attack by submitting the same ID token that does not contain any additional abbitruries or features such encrypted nonce or encrypted timestamps or any additional security functions on the OpenID Connect request. While  the proposed solution that we have proposed is to apply nonce in the OpenID Connect request  and also to add a security function that was suggested by Yousra et al. (2023) which is called  “owner of()” that is related to the ID token that checks whether the source of the ID token is same source of the ID token owner or not in order to improve the authentication security.

# Results and Discussion

## Explanation of the Programs

The programs I developed  work together to simulate a secure authentication process using the OpenID Connect Protocol. I incorporated various security enhancements to mitigate replay attacks and unauthorized access attempts as we proposed above. The client interacts with the server to obtain authentication tokens and access protected resources, while the server validates requests and ensures secure communication between parties. Below I am illustrating each program’s steps how it works:

## Client Program

1. The client script serves as a simulated client application that interacts with the OpenID Provider (OP) and the relying party (RP) to authenticate users and access resources securely.
2. It initializes with the necessary parameters such as the OP URL, RP URL, username, password, and encryption key.
3. The OpenIDConnect class encapsulates the functionality required for authentication and resource access.
4. Upon initialization, it sets up the necessary attributes and encryption key for secure communication.
5. The request\_identity\_token method sends a request to the OP to authenticate the user and obtain an identity token.
6. the send\_data\_to\_server method encrypts the nonce and sends it to the server for validation.
7. Upon successful authentication, the identity token, session ID, and nonce are received and stored.
8. The request\_access\_token method sends a request to the OP to obtain an access token using the received identity token.
9. The access\_resource method accesses a protected resource at the relying party using the obtained access token.

## Server Program

1. The server script represents the backend server responsible for handling authentication requests, issuing tokens, and granting access to resources.
2. It initializes a Flask application to define the server routes and handle incoming requests.
3. The server includes routes for user authentication, token issuance, resource access, and nonce validation.
4. Upon receiving authentication requests (/openid/authenticate), it validates the user's credentials and generates a session ID and encrypted nonce for the authenticated user.
5. The nonce is encrypted using the encryption key before being sent back to the client.
6. The server also includes a separate endpoint (/openid/authen) to receive and validate encrypted nonces sent by the client.
7. The server route for token issuance (/openid/token) validates the received identity token and issues an access token with a reduced lifespan.
8. To access protected resources (/rp/resource), clients must provide a valid access token in the request headers.
9. Enhanced logging is implemented throughout the server script to provide detailed information about each step of the authentication process and security measures applied.

A computer screen with white text

Description automatically generated

***Figure5 .*** The Server is running



***Figure6 .*** Client Connection

Client started a connection and logged in with his username and passwd, the server initiated for him a session and a session ID.

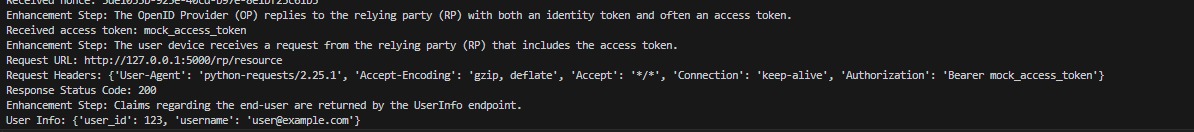


***Figure7.*** Received Identity token, nonce, session ID



***Figure8.*** Received Nonce

The received nonce is encrypted and the client will decrypt it with the shared key , attach a message and send it to the server to authenticate him.



***Figure9.*** Done Authentication

The authentication is done; user authenticated and has been provided with a token ( its lifetime is short ) and he can access the resources ( info such as id and username) .

# Pseudocode

Define a class named OpenIDConnect

Constructor:

Initialize the class with the OpenID Provider URL (op\_url), Relying Party URL (rp\_url), username, and password

Initialize access\_token, identity\_token, session\_id, nonce, and my\_key

Method user\_sign\_in:

Print a message that indicates that the user is signing in

Method request\_identity\_token:

Send a GET request to op\_url/authenticate with username and password parameters

If the response status code is 200:

Extract identity\_token, session\_id, and nonce from the response

Else:

Print a failure message

Method request\_access\_token:

If identity\_token is available:

Send a POST request to op\_url/token with identity\_token

If the response status code is 200:

Extract access\_token from the response

Else:

Print a failure message

Else:

Print a message that indicates that the identity\_token is not available

Method access\_resource:

If access\_token is available:

Send a GET request to rp\_url/resource with Authorization header containing access\_token

If the response status code is 200:

Extract user\_info from the response

Else:

Print a failure message

Else:

Print a message that indicates that the access\_token is not available

Method send\_data\_to\_server(data):

Decrypt the nonce using my\_key

Encrypt the decrypted nonce

Send a POST request to op\_url/authen with the encrypted nonce

If the response status code is 200:

Print a success message

Else:

Print a failure message

Instantiate the OpenIDConnect object with op\_url, rp\_url, username, and password

Call user\_sign\_in method

Call request\_identity\_token method

Call request\_access\_token method

Call access\_resource method

Call send\_data\_to\_server method with some data

## Expected Results  ( Enhancing Security Measure)

1. Identity Access Management (IAM) Integration:
   1. IAM integration,  enhances authorization mechanisms and ensures that only authorized parties gain restricted access to user resources securely.
   2. Expected Result: Users can grant limited access to their resources without revealing their credentials, thus minimizing the risk of unauthorized access by malicious entities.
2. Reduced ID Token Expiration Time:
   1. Shortening the expiration time of ID tokens from 120 minutes to 15 minutes mitigates the risk of replay attacks by limiting the validity period of tokens.
   2. Expected Result: ID tokens have a shorter lifespan, reducing the window of opportunity for attackers to exploit stolen or intercepted tokens.
3. Cryptographic Nonce Generation and Authentication:
   1. Utilizing cryptographic functions for encrypted nonce generation and authentication ensures that each request is authentic and originated from a legitimate user.
   2. Expected Result: Nonces are encrypted before transmission, preventing interception and tampering by malicious actors.
4. Use of security authentication functions, such as verifying the source of ID tokens, further enhances security by preventing replay attacks.
5. Improved Traceability and Logging:
   1. Enhanced logging provides detailed information about each step of the authentication process and the security measures applied.
   2. Expected Result: Logs capture information such as user authentication attempts, token issuance, resource access, and nonce validation, facilitating auditing and troubleshooting processes.
6. Efficient Nonce Validation:
   1. Nonce validation mechanisms ensure that only legitimate nonces are accepted, preventing replay attacks and unauthorized access attempts.
   2. Expected Result: Nonces sent by the client are validated against stored nonces on the server, and only valid nonces are accepted for further processing.
7. Reduced Vulnerability to Replay Attacks:
   1. Implementation of security enhancements reduces the vulnerability of the OpenID Connect Protocol to replay attacks, enhancing overall system security.
   2. Expected Result: Replay attacks are mitigated through the adoption of IAM integration, reduced token lifespan, and cryptographic nonce generation and authentication.

These changes should make logging in safer by making it harder for attackers to replay login attempts. We'll also have better records of who's logging in and when, and communication between the client and server will be more secure. These upgrades are meant to make the OpenID Connect Protocol more secure overall, protecting user accounts and data from unauthorized access.

## Comparison with Yousra et al. (2023)

Enhancements for Security: Both my work and the paper by Yousra et al. aim to enhance the security of the OpenID Connect Protocol. While my work focuses on implementing enhancements such as multi-factor authentication, reduced token lifespan, and cryptographic nonce generation, Yousra et al. propose a Blockchain-based OIDC approach to enhance data security, integrity, and privacy protection. I saw my work didn’t meet with Yousra in performance evaluation , she ensured that the proposed enhancements do not significantly affect user experience or the performance of existing systems, while I had used more storages than it’s required, I made it somehow harder for the user to login each time by reducing the lifetime of the token. We both look at the importance of secure authentication mechanisms in systems. While my work implements enhancements within the existing OpenID Connect Protocol, Yousra et al. propose a novel approach leveraging blockchain technology for authentication and data protection.

Scope of Enhancements: My work primarily focuses on enhancing authentication security through measures like MFA, reduced token lifespan, and cryptographic nonce generation. On the other hand, Yousra et al. propose a comprehensive model that addresses data security, integrity, availability, revocation, and delegation using blockchain technology.

Formal Security Analysis: Yousra et al. conduct a formal security analysis of their proposed system using tools like AVISPA and Scyther to verify its security. In contrast, my work focuses more on practical implementation of python programming language.

## Comparison with Ofleh (2018)

Focus on OpenID Connect Protocol: Both my work and Ofleh (2018) focus on the OpenID Connect Protocol and its role in identity and access management (IAM). Ofleh discusses the importance of robust IAM implementations, while my work implements security enhancements as we proposed solutions above. The difference I saw that Ofleh suggests proposals like OpenID certification and two-factor authentication (2FA), while my work implements measures such as MFA, reduced token lifespan, and cryptographic nonce generation. I saw Ofleh discusses recommendations for improving the OpenID Connect Protocol without focusing on specific technological implementations. In contrast, my work implements practical enhancements using encryption and secure communication mechanisms.

Scope of enhancements: Ofleh discusses a broader range of recommendations for improving the OpenID Connect Protocol, including aspects such as identity ownership, verification, and data access monitoring. My work primarily focuses on security enhancements related to authentication mechanisms, as I developed a program that simulates the same exact protocol, then I compared the old program with the new ehanced program.

# Future Work

While this study contributes valuable insights into how to improve the open-connect protocol and proposes many solutions, several areas need further exploration. In our paper, we focused on resolving mostly the token replay attack and introduced new mechanisms like multi-factor authentication (MFA) and identity access mechanism (IAM) to a protocol that is widely connected with single-sign-on (SSO).

We also proposed a reduction in the time of the token ID and added a function to prove the source from which the token is coming to prevent the replay attack. These are changes that are sufficient and serve the need, but if we delve deeper into the protocol, we will come across another vulnerability that is faced, which is ID spoofing, which involves an attacker impersonating a malicious IdP and producing tokens in the name of other trusted IdPs.

 Even though the OpenID Foundation offers a certification process for organizations to declare the accuracy of their implementation of the OpenID Connect protocol, this process itself doesn't directly prevent attacks like ID spoofing. We could explore how to have strict token validation by ensuring that the identity provider (IdP) issues tokens with accurate and verifiable information. Service Providers (SPs) should validate tokens thoroughly, checking the issuer ('iss') and subject ('sub') claims to ensure they match the expected values and verifying the signature of ID tokens to ensure their integrity and authenticity(Fries,2020).

Also, we should stay away from relying on fields like name or email, as these can be played with easily. We propose to use strong cryptographic algorithms to properly manage keys and tokens, as this will increase the security of this mechanism. But with such implementations, another question arises, which is: how does that affect the performance of the system? And what is the algorithm that should be used? Each scenario has its tradeoffs, and we need to focus on what exactly our goal is and the context we are in.

Given the widespread use of OpenID in everyday online interactions, it is important that performance considerations are weighed alongside the necessity of basic security measures. While preventing attacks is important, the proposed modifications are intended not only to strengthen the security defenses of OpenID systems but also to ensure that the user experience remains seamless .The challenge is to achieve a balance where enhanced security protocols work without reducing the system performance, which will maintain the seamless and efficient nature of OpenID Connect as a reliable and easy-to-use solution.

# 10.Conclusion

In this paper, we explored the concepts of identity access management (IAM) and multi-factor authentication (MFA) in the context of the Zero Trust Model, recognizing the critical roles these mechanisms play in modern cybersecurity architectures. A key component of cybersecurity, authentication ensures the authenticity of individuals, systems, or devices requesting to access digital resources.

Our contribution addresses a critical gap in existing literature by combining MFA and IAM within the Zero Trust Model, strengthening security measures for organizations. After conducting a comprehensive review and comparative analysis of research papers, we identified the positive aspects, limitations, and strategic implications of integrating MFA and IAM.

We provide recommendations to strengthen authentication systems in the Zero Trust Model based on our observations. These suggestions support the use of biometric data to strengthen security, the adoption of complete MFA and IAM solutions, and the integration of edge computing and attribute-based access control, particularly in sensitive settings like public institutions and smart cities.

Our research concludes by proposing some recommendations and a solution based on what we observed and found in the literature review papers to improve the authentication process and make it more secure in cloud computing within the zero trust model. Our solution's focus was to improve the OpenID Connect protocol (OIDC) by integrating it with identity access management and multi-factor authentication.

This paper resulted in our implementation of a solution that mitigated replay attacks by replacing single sign-on (SSO) with multifactor authentication (MFA) and combining it with identity and access management (IAM) in the OIDC protocol within the cloud computing zero trust model to improve the authentication process.

We also planned future work to expand our solution and implement some mechanisms that mitigate ID spoofing (IDS) attacks that target an OpenID protocol vulnerability related to identity authentication in Single Sign-On (SSO) systems.

This comprehensive strategy for access control and authentication within the Zero Trust Model highlights the importance of multi-layered security measures to prevent cyberthreats. Organizations may strengthen their security defenses and ensure strong protection of their systems in a constantly changing threat landscape by implementing our recommendations.

# Members contribution

|  |  |
| --- | --- |
| **Contributors** | **Sections Contributed** |
| **Afra Mustafa** | abstract, methodology section, problem section, introduction to MFA, summarizing two papers in the literature review section, comparative table analysis, motivation of the proposed solution, proposed solution explanation (with Flowchart), and innovation journey section. |
| **Jude Abdel Halim** | Introduction to IAM, importance of authentication, summarizing one paper in literature review section, structure section, comparative table analysis, modified abstract, pseudocode, and report structure. |
| **Kassem Darawcha** | First paragraph of the abstract, problem section, summarizing one paper in literature review section, comparative table analysis, observation section, results and discussion, expected results, comparison with literature review. |
| **Muaz Osman** | Third paragraph of the abstract, summarizing two papers in the literature review section, comparative table analysis, recommendation section, proposed solution introduction, and ID spoofing attack. |
| **Rana Alzakary** | Introduction to Zero Trust Model, contribution section, motivation section, summarizing three papers in literature review section, comparative table analysis, conclusion, and replay attack. |

# References

Amazon Web Services. (2024). *What is IAM?*. AWS identity and access management. <https://docs.aws.amazon.com/IAM/latest/UserGuide/introduction.html>

Auth0. (2023). *Mitigate replay attacks when using the implicit flow*. Auth0 Docs. <https://auth0.com/docs/get-started/authentication-and-authorization-flow/implicit-flow-with-form-post/mitigate-replay-attacks-when-using-the-implicit-flow>

Barney, N., Shacklett, M. E., & Rosencrance, L. (2023, November 1). *What is authentication?: Definition from TechTarget*. Security. <https://www.techtarget.com/searchsecurity/definition/authentication>

Chen, L., Sun, Y., & Sun, Z. (2021). A Mobile Internet Multi-level Two-way Identity Authentication Scheme Based on Zero Trust. *2021 IEEE 23rd Int Conf on High Performance Computing & Communications; 7th Int Conf on Data Science & Systems; 19th Int Conf on Smart City; 7th Int Conf on Dependability in Sensor, Cloud & Big Data Systems & Application (HPCC/DSS/SmartCity/DependSys)*. <https://doi.org/10.1109/hpcc-dss-smartcity-dependsys53884.2021.00243>

CISA. (2022, January 5). *Multi-factor authentication (MFA): CISA*. Cybersecurity and Infrastructure Security Agency CISA. <https://www.cisa.gov/resources-tools/resources/multi-factor-authentication-mfa>

Cisco. (2024, February 8). *What is multi-factor authentication (MFA)?*. Cisco. <https://www.cisco.com/c/en/us/products/security/what-is-multi-factor-authentication.html>

CyberArk. (2024). Configuring the token time stamp. <https://www.ibm.com/docs/en/sva/8.0.1?topic=signon-configuring-token-time-stamp>

Daah, C., Qureshi, A., Awan, I., & Konur, S. (2024). Enhancing Zero Trust Models in the Financial Industry through Blockchain Integration: A Proposed Framework.*Electronics, 13*(5), 865. <https://doi.org/10.3390/electronics13050865>

Fett, D., Kusters, R., & Schmitz, G. (2017). The web SSO standard openid connect: In-depth formal security analysis and security guidelines. *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*. <https://doi.org/10.1109/csf.2017.20>

Fries, C.(2020). Security Analysis of Real-Life OpenID Connect Implementations.

Garbis, J., Chapman, J.W. (2021). Identity and Access Management. In: Zero Trust Security. *Apress, Berkeley, CA*. 71–9. <https://doi.org/10.1007/978-1-4842-6702-8_5>

Han, A. H., & Lee, D. H. (2023, October 5). Detecting Risky Authentication Using the OpenID Connect Token Exchange Time. *Sensors*, *23*(19), 8256. <https://doi.org/10.3390/s23198256>

Heilman, E., Mugnier, L., Filippidis, A., Goldberg, S., Lipman, S., Marcus, Y., Milano, M., Premkumar, S., Unrein, C., & Merfeld, J. (2024, April 3). *OpenPubkey: Augmenting OpenID Connect with User held Signing Keys*. IACR Cryptology ePrint Archive. <https://ia.cr/2023/296>

*IBM Cloud Application Performance Management, Private 8.1.4*. (n.d.). <https://www.ibm.com/docs/en/capmp/8.1.4?topic=administering-adjusting-console-timeout-value>

IBM. (2024). What is identity and Access Management (IAM)? <https://www.ibm.com/topics/identity-access-management>

Iordache, C. A., Dragomir, A. V., & Marian, C. V. (2022). Public Institutions Updated Enhanced Biometric Security, Zero Trust Architecture and Multi-Factor Authentification. *International Symposium on Electronics and Telecommunications (ISETC)*. 1-4 <https://doi.org/10.1109/isetc56213.2022.10010127>

Kanade, V. (2023, December 4). *Authentication meaning, types, and tools: Spiceworks*. Spiceworks. <https://www.spiceworks.com/it-security/identity-access-management/articles/what-is-authentication/>

KOSE, B. O., BUK, O., MANTAR, H. A., & COSKUN, V. (2020). TrustedID: An Identity Management System based on OpenID Connect Protocol. *2020 4th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, 1–6. <https://doi.org/10.1109/ismsit50672.2020.9254886>

Kovacevic, I., Stojkov, M., & Simic, M. (2024). Authentication and Identity Management based on Zero Trust Security Model in micro-cloud environment. *Disruptive Information Technologies for a Smart Society*, *872*, 481–489. <https://doi.org/10.1007/978-3-031-50755-7_45>

Microsoft . (2024). *What is openid connect (OIDC)?*. Microsoft Security. <https://www.microsoft.com/en-us/security/business/security-101/what-is-openid-connect-oidc>

Microsoft security. (2024). *What is Identity Access Management (IAM)?*. What is Identity Access Management (IAM)? | Microsoft Security. <https://www.microsoft.com/en-us/security/business/security-101/what-is-identity-access-management-iam>

Nguyen, H. H., Lim, Y., Seo, M., Jung, Y., Kim, M., & Park, W. (2023). Strengthening Information Security Through zero trust architecture: A case study in South Korea. *Communications in Computer and Information Science*, *1950*, 63–77. <https://doi.org/10.1007/978-981-99-7666-9_6>

Ofleh, O. (2018). Future of Identity and Access Management: The OpenID Connect Protocol (Doctoral dissertation).‏

Ofleh, O. (2018). *Future of Identity and Access Management: The OpenID Connect Protocol* (Doctoral dissertation). <http://hdl.handle.net/10657/3381>

Okta. (2024). *OpenID connect protocol*. Auth0 . <https://auth0.com/docs/authenticate/protocols/openid-connect-protocol>

OpenID Foundation. (2023). How OpenID Connect works. <https://openid.net/developers/how-connect-works/>

Quest. (2024). *What is strong authentication?*. One Identity. <https://www.oneidentity.com/what-is-strong-authentication-in-cybersecurity/>

Said, W., Mostafa, E., Hassan, M. M., & Ayman, M. M. (2022). A Multi-Factor Authentication-Based Framework for Identity Management in Cloud Applications. *Computers, Materials, & Continua, 71*(2), 3193-3209. <https://doi.org/10.32604/cmc.2022.023554>

Shore, M., Zeadally, S., & Keshariya, A. (2021, November). Zero Trust: The What, How, Why, and When. *Computer*, *54*(11), 26–35. <https://doi.org/10.1109/mc.2021.3090018>

Yousra, B., Yassine, S., Yassine, M., Said, S., Lo’ai, T., & Salah, K. (2023). A Novel Secure and Privacy-Preserving Model for OpenID Connect Based on Blockchain. IEEE Access, 11, 67660–67678. <https://doi.org/10.1109/access.2023.3292143>

Yousra, B., Yassine, S., Yassine, M., Said, S., Lo’ai, T., & Salah, K. (2023). A Novel Secure and Privacy-Preserving Model for OpenID Connect Based on Blockchain. *IEEE Access*, *11*, 67660–67678. <https://doi.org/10.1109/access.2023.3292143>

Yousra, B., Yassine, S., Yassine, M., Said, S., Lo’ai, T., & Salah, K. (2023). A novel secure and privacy-preserving model for OpenID connect based on Blockchain. *IEEE Access*, *11*, 67660–67678. <https://doi.org/10.1109/access.2023.3292143>

*Zero Trust Model - Modern Security Architecture | Microsoft Security*. (n.d.). <https://www.microsoft.com/en-us/security/business/zero-trust>